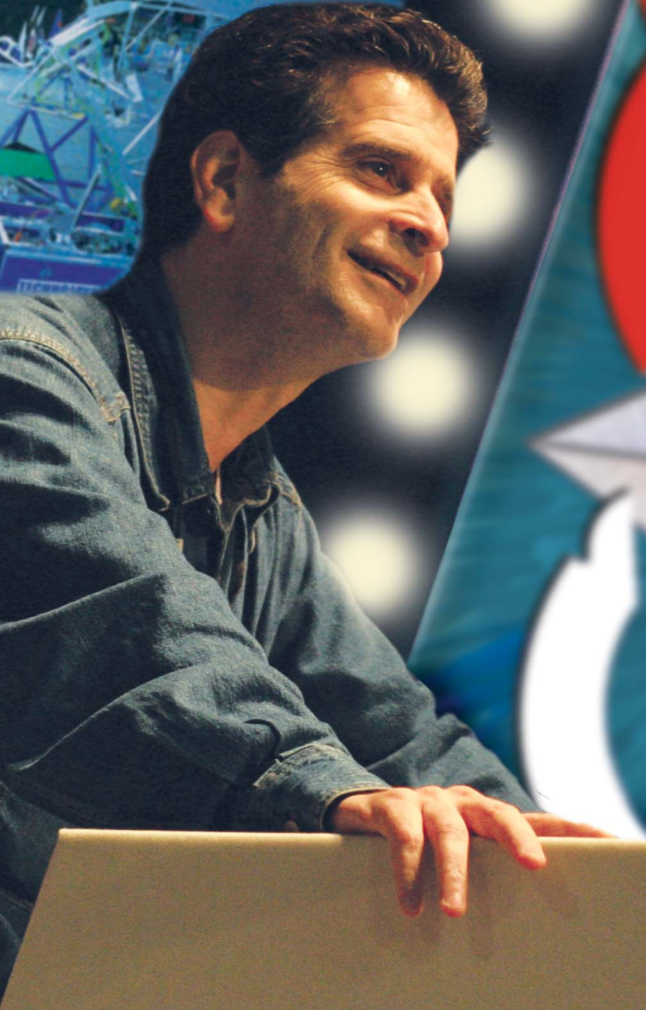
WHY HUMANOIDS • ALONG CAME BOBBY • DARPA REVISITED

# SERVO MAGAZINE

For the Next Generation of Robotics Experimenters

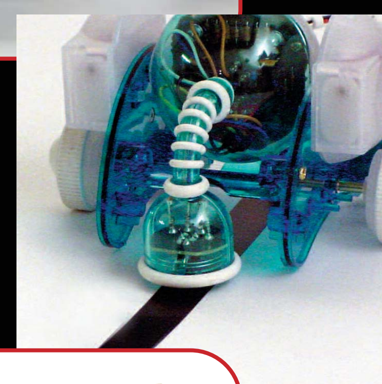
## FIRST *at Last!*

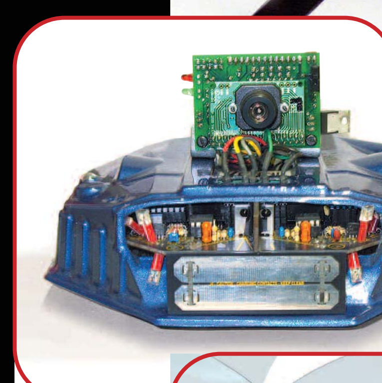
### 2006 Season Kicks Off With Over 1,100 Teams

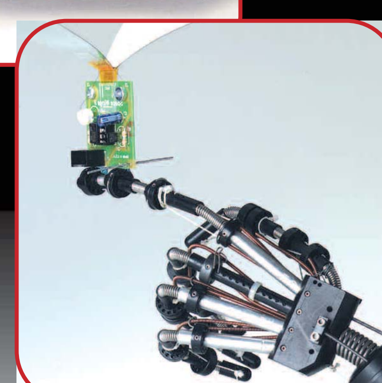


- STEPPING MOTOR BASICS
- PROGRAMMABLE LOGIC
- BUILD A LAPTOP- OR PDA-BASED ROBOT

www.servomagazine.com







ROBOTICS HITEC

# Assimilate This...

## Hitec Robotics introduces ROBONOVA-I

The remarkable ROBONOVA-1 is an exciting humanoid robot offering educators, students and robotic hobbyists a complete and ultra modern robot package.

The kit contains a detailed English manual and all components necessary to build, program and operate your own robot.

- Remocon IR Controller
- Sixteen HSR-8498HB servos
- MR-C3024 Micro controller board
- Re-chargeable NiMH battery and charger
- Robo Basic programming software and manual
- PC Serial port programming interface cable


### MR-C3024 - Robot Controller (#78001)

Using the latest mega 128 MPU controller from ATMEL, the stable ROBONOVA-1 can walk, do flips, cartwheels, dance moves, and is ready to compete in any Robo One class "J" competition.

### HSR-8498HB - Hitec Digital Robot Servo (#38498)

This fully articulating, 12" high, mechanical man is controlled with 16 powerful Hitec HSR-8498HB digital servos built specifically for the ROBONOVA-1. These custom servos feature super strong Karbonite gear trains and "feedback" technology for easy programming.

## ROBONOVA-I

For more information on availability and pricing, call us at 858-748-6948 or visit, **www.hitecrobotics.com**

Available as a kit [**#77000**]
or pre-assembled, RTW [Ready To Walk] package [**#77002**]

# SERVO

## Features & Projects




## ON THE COVER

FIRST (For Inspiration and Recognition of Science and Technology) is a multinational non-profit organization, that aspires to transform culture, making science, math, engineering, and technology as cool for kids as sports are today.

FIRST was founded in 1989 by Dean Kamen, inventor of the Segway Human Transporter (shown in lower left corner). Dr. Woody Flowers (shown in the upper right) is the co-founder and vice-chairman of FIRST.

FIRST operates the FIRST Robotics Competition in which teams of high school students, sponsored and assisted by local companies and volunteers, design, assemble, and test a robot capable of performing a specified task in competition with other teams. FIRST also runs the FIRST LEGO League, for children 9-14 years old, and FIRST Place, an innovative science and technology center, including a hands-on children's science museum.

To get all the details, events, and ways to get involved in FIRST, go to **www.usfirst.org**


*SERVO Magazine* **(ISSN 1546-0592/CDN Pub Agree#40702530)** is published monthly for $24.95 per year by T & L Publications, Inc., 430 Princeland Court, Corona, CA 92879. APPLICATION TO MAIL AT PERIODICALS POSTAGE RATE IS PENDING AT CORONA, CA AND AT ADDITIONAL ENTRY MAILING OFFICES. POSTMASTER: Send address changes to **SERVO Magazine, P.O. Box 15277, North Hollywood, CA 91615** or Station A, P.O. Box 54, Windsor ON N9A 6J5; cpcreturns@servomagazine.com

# 01.2006

VOL. 4 NO. 1

## Departments

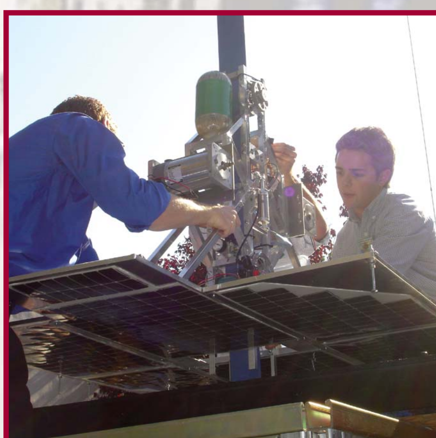


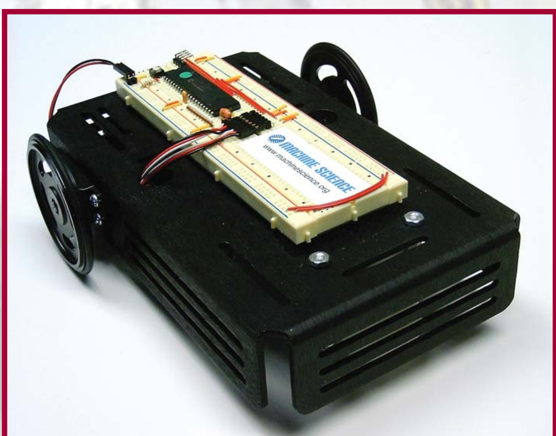
## Columns



## Coming 02.2006

Space Elevator — Building a Highway to the Stars






Using Interrupts to Control Servos

# Mind / Iron

by Dave Prochnow

Mark Tilden's done it. Dave Hrynkiw's done it. Gareth Branwyn's done it. And Ben Wirz has done it. Each of these noted robotics physicists has designed a successful introductory robot. Unfortunately, none of these robot builders has succeeded in creating a viable robot kit for kids. Specifically, a robot kit for kids that is not kid-like in its design. Before I get ahead of myself, let me provide you with a little background information.

I had the distinct pleasure of hosting a program at RoboNexus 2005 in San Jose, CA that instructed Bay Area youngsters how to build their first robot kit. At the behest of *SERVO Magazine*, I manned the "Kit Kollege" booth. Along with my ably bodied assistant — my oldest daughter, Katherine — we helped over 160 participants build an OWI Roly-Poly robot kit.

My goal was to educate the students about the future of robotics and, in particular, the importance that robotics play in our daily lives all wrapped up inside a simple, easy-to-build robot kit-building demonstration. Generous donations of kits, tools, and batteries helped to cement this goal into a wildly successful program.

During the three-day event, Katherine and I helped over 160 youngsters (of all ages) assemble the kits and get "jazzed" about robots. In acknowledgment of this effort, we were recognized by the event organizer and the local area teacher's association for being one of the most "appealing and appreciated demonstrations" at RoboNexus 2005.

All was not bliss in my presentation, however. While this three-day event was an eye-opening experience, my program also served to highlight three major shortcomings in teaching an introductory robot kit building program:

**1.** *Cost*. Although the kits were donated by a robot kit manufacturer, the supply was limited. Additionally, when the kit supply was exhausted only a small handful of extra robot kits could be purchased locally for supplementing this program (i.e., an availability issue). Robot kits that cost even as little as $10-$20 each are too expensive for this type of program.

*Lesson Learned*: Cost should never be a reason for turning away budding robotics physicists.

**2.** *Tools*. The selected robot kit could not be assembled without a couple of simple hand tools. Similar to the kits themselves, a generous supply of tool kits was donated by a local electronics company, but the total number provided did not match the number of kits. Therefore, a substitute tool kit was hastily provided to the participants, but this second tool kit did not contain all of the required tools.

*Lesson Learned*: Tools should always be supplied with robot kits; or, ideally, a kit should not require any tools for assembly.

**3.** *Batteries*. One of the greatest pollutants to our world's landfills is the "disposable" dry cell battery. While an ample supply of batteries

**Mind/Iron Continued** →

Published Monthly By
**T & L Publications, Inc.**
430 Princeland Court
Corona, CA 92879-1300
**(951) 371-8497**
FAX **(951) 371-3052**
Product Order Line **1-800-783-4624**
**www.servomagazine.com**

Subscriptions
Inside US **1-877-525-2539**
Outside US **1-818-487-4545**
P.O. Box 15277
North Hollywood, CA 91615

**PUBLISHER**
Larry Lemieux
**publisher@servomagazine.com**

**ASSOCIATE PUBLISHER/ VP OF SALES/MARKETING**
Robin Lemieux
**display@servomagazine.com**

**CIRCULATION DIRECTOR**
Tracy Kerley
**subscribe@servomagazine.com**

**WEB CONTENT/STORE**
Michael Kaudze
**sales@servomagazine.com**

**PRODUCTION/GRAPHICS**
Shannon Lemieux
Michele Durant

# BIO-FEEDBACK

*Dear SERVO:*

I enjoy my *SERVO* and *Nuts & Volts* magazines and purchase electronic components and kits from some of your advertisers. I do have one problem with some of the stores.

I live in Alaska and shipping charges can get expensive. It seems odd that I can order parts from companies overseas and get them shipped to Alaska in only 10 days for $7.50 for orders under $50. A few of the US stores won't ship by the US Postal Service and only use UPS for sending items out. This means for a similar order to Alaska, they want $21 or more for shipping and it still takes about four or five days. Other stores will ship via the Postal Service, and on small orders they usually use Priority Mail, which costs about the same as ordering from overseas, but I get items in about three or four days.

In fact, the USPS has two flat rate Priority Mail boxes that have no weight limit and the postage is $7.70. The Postal Service will pick up these boxes from the shipper and they are willing to provide tracking for only a small extra fee. No I don't work for the post office. Maybe if you print this letter, other companies might reconsider using Priority Mail for the residents of Alaska and Hawaii so we can afford our electronics hobby.

**Jeb Stuart**


was available for the donated kits, the supplementary-purchased robot kits had a vastly different battery requirement. We were unable to meet this other battery requirement and, therefore, the participants were not able to test their robots during the program. Introductory robot kits should never use batteries for power.

*Lesson Learned*: A simple, inexpensive solar cell power plant is much more practical, as well as being less polluting to our future.

Following my return from RoboNexus 2005, I set out to find a robot kit that would eliminate these three obstacles while providing a low-cost, painless introduction to robotics. My mantra became, "there's got to be a better way."

Unfortunately, almost every "low-cost" introduction to robotics costs an arm and a leg ... literally. Therefore, the students who would most benefit from a simple introductory robot kit are excluded from this education based on the economics of today's overly complex digital robot designs. There's got to be a better way.

This quest for a low-cost painless introductory robot kit is not new. Science teachers around the world have been begging robot manufacturers to invent just such a critter. The results have always been less than satisfying. Much like the educational computer myth, typically, educators are left with a pile of robot refuse that is short on usability but long on costly support. There's got to be a better way.

Do you have a better way? If so, *SERVO Magazine* is the perfect storefront for matching your entrepreneurial spirit to a customer base of schools, clubs, and organizations who are hungry for a better way in beginner robot kit designs.

Before you attempt to recycle that old solar-powered love bug kit design, however, remember that a robot kit for kids shouldn't be kid-like in its design. Kids are savvy shoppers. Give 'em what they want. Your design must be intelligent, expandable, and powerful. So get out your schematic drafting tools, clean out your parts bins, and create a better introductory robot kit.

Don't wait, though. RoboNexus 2006 is right around the corner and your robot kit could be the stimulus that creates the next Mark W. Tilden. SV

# Robytes

by Jeff Eckert

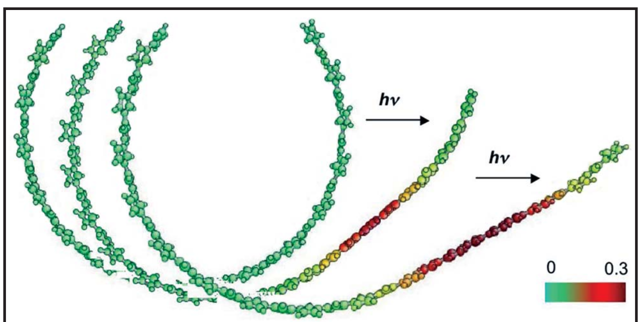
**Are you an avid Internet surfer who came across something cool that we all need to see? Are you on an interesting R&D group and want to share what you're developing? Then send me an email! To submit related press releases and news items, please visit www.jkeckert.com**

***— Jeff Eckert***

## Bionic Speed in Artificial Muscles?

Electroactive polymer actuators have been used for some time as artificial muscles and have promise as an alternative to shape memory alloys, magnetostrictive materials, and so on. When you send the right kind of electrical energy (a charge density wave or "soliton") into a conductive polymer, it will actuate (contract or expand), thus accomplishing some mechanical purpose.

A major drawback, however, is that they react about 100 times more slowly than human muscles. That is because the polymers are traditionally doped with ions that expand the volume of the material and give it strength, which also makes them heavy and slow. But researchers at MIT (**www.mit.edu**) have now theorized that the doping is unnecessary, and it is possible to activate the polymer material with a particular frequency of light. Such a method would allow the polymer to react much more quickly — as much as 1,000 times as fast as human muscles. The concept appears to be all theoretical at this point, but research continues.




***New polymer material could allow a robotic muscle to be actuated with light. This polymer chain is naturally curved before exposure but straightens where light creates positive charges (red area). Photo courtesy of Yip lab, MIT.***

## World's Smallest Bot

***Dartmouth's microbot is untethered and controllable, yet 200 of them could fit in line across the top of an M&M. Photo courtesy of the Donald Laboratory.***

Researchers at Dartmouth College (**www.dartmouth.edu**) have created what they claim is the world's smallest robot, with dimensions of only 60 by 250 µm. It integrates power delivery, locomotion, communication, and a controllable steering system into the package, which has never before been accomplished in so small a machine. According to Prof. Bruce Donald, "It's tens of times smaller in length, and thousands of times smaller in mass than previous untethered microrobots that are controllable. When we say 'controllable,' it means it's like a car; you can steer it anywhere on a flat surface and drive it wherever you want to go. It doesn't drive on wheels but crawls like a silicon inchworm, making tens of thousands of 10 nm steps every second. It turns by putting a silicon 'foot' out and pivoting like a motorcyclist skidding around a tight turn."

The robot contains two independent microactuators: one for forward motion and one for turning. It's not preprogrammed; rather, it is teleoperated, powered by a grid of electrodes upon which it walks. The charge in the electrodes not only provides power, it supplies the robot with instructions that allow it to move freely over the electrodes, unattached to them. There doesn't appear to be any specific purpose for the device at this time, but MIT cites IC inspection and repair and biotechnology among possible applications. In the meantime, it sure is small.

## Chimp Head Apes the Real Thing

***This life-size chimp head mimics the real thing. Photo courtesy of Sharper Image.***

If you sort-of miss your ex-husband but don't really want him back, take a look at the "Alive" Chimpanzee from the Sharper Image (**www.sharperimage.com**). Not only is it realistic looking, his head, neck, face, and eyes move. The eyes track movements in the room using infrared vision, his ears have sensors for stereoscopic sound detection, and his skin reacts to contact with touch sensors.

The chimp is capable of four emotional states: curious, happy, fearful, and feisty. He can mimic the sounds of a live chimp, reflecting his emotions with various screeches and whoops. And perhaps best of all, you can override the autonomous functions via a wireless control. The unit runs on four D cells or the AC adapter and sells for $149. That's only a third of what you paid for that silly Ionic Breeze, and a lot more entertaining.

## Exoskeleton System Developed

*The HAL exoskeleton power assist system, worn by Prof. Sankai. Photo courtesy of Cyberdyne, Inc.*

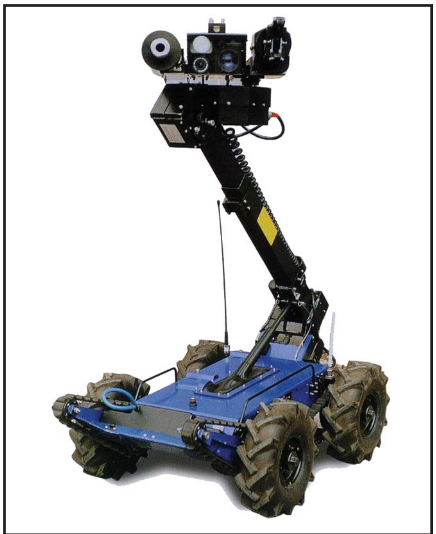
An interesting area of robotics technology comes in the form of exoskeletal devices attached to a real human being, and one that appears to be nearing the practical application stage is the Hybrid Assistive Limb (HAL) system from Cyberdyne, Inc. (**www.cyberdyne.jp**). As of this writing, it doesn't appear that HAL is commercially available, but it has been demonstrated at symposia and is supposed to go into production soon.

Various sources have speculated that the device will be priced at about $14,000. Developed via research at the University of Tsukuba under the direction of Prof. Yoshiyuki Sankai, who is also with Cyberdyne, it is intended to provide walking assistance to people who have gait disorders. The present model, HAL-5, employs a range of angle sensors, myoelectrical sensors, floor sensors, and so on to determine the state of itself and the operator, and a backpack contains motor drivers, the computer and measurement system, a wireless LAN, and the power supply.

The end result is a device that provides autonomous control and power assist to the wearer, based on biological feedback and predictive feedforward. Although intended for elderly and disabled people, it could also have wider uses, e.g., giving added strength to firefighters, furniture movers, and soldiers. And if you could strap your mother-in-law into a remote-controlled version and grab the joystick, wonderful things could happen.

## Robot Saves Tweety

It costs $120,000 and was designed for improvised explosive device disposal, conventional munitions disposal, and other high-risk tasks, but the Cyclops miniature remotely operated vehicle (MROV) from AB Precision (**www.abprecision.co.uk**) recently demonstrated its utility for cockatiel retrieval. Apparently, the ground opened up beneath an apartment building in Sydney, Australia, and residents were required to evacuate immediately. It was deemed too dangerous to allow human beings back in, but the police rescue squad sent in the Cyclops, which snagged Tweety the cockatiel's cage with its extendable mast and brought him to safety. As last report, Tweety was fine and in good spirits. (Everyone say, "awwww.")



*The Cyclops MROV from AB Precision. Photo courtesy of ABP.*

## New Comic Strip Initiated

Inspired by Tony Cheetham's established "Only Human" comic strip, the "swirling brain" of Jim Brown has created a new strip, called Emergent Behavior. It is based on characters who are plotting to get robotics research grants, and you can expect unpredictable behavior by both humans and robots. To get a regular dose of both strips, just visit **robots.net/comics/** **SV**


*The new Emergent Behavior comic strip by Jim Brown has debuted.*

Our resident expert on all things robotic is merely an Email away.
**roboto@servomagazine.com**

**Tap into the sum of *all human knowledge* and get your questions answered here! From software algorithms to material selection, Mr. Roboto strives to meet you where you are — and what more would you expect from a complex service droid?**

# ASK MR. ROBOTO

by
**Pete Miles**

**Q**. I have noticed that you show a lot of different types of electronic circuits in your articles. I have a 150-in-1 electronic projects lab from RadioShack, but I can't make any of your projects with it. What do you use to make these circuits?

**— Bill Mease**

**A**. Every single one of my projects starts out on what is called a "solderless breadboard." Solderless breadboards work in much the same manner as the 150-in-1 electronics project lab that you have, where you plug in wires from one component to another to make an electrical circuit. What is different is that with the solderless breadboard, you can plug in your own choice of electronic components, such as a BASIC Stamp microcontroller.

Figure 1 shows an electrical illustration of how these boards work. The board is broken up into two groups of five hole columns. All five holes in a single column are connected together (shown by the green line connecting all five boxes) so that any wires plugged in the same column are electrically connected together. The gap between the two groups of five hole columns is wide enough so that a standard integrated circuit can be plugged in across the gap so that there will be four holes available for connecting other wires and components to each lead of the IC.

Small solderless breadboard can have as few as 170 connection holes (17 columns of two five hole groups) to large scale boards with over 3,000 holes. They come in many different sizes and configurations, and they are fairly inexpensive. They can also be found at most electronic stores. Just ask the sales representative where the solderless breadboards are.

As to the question as to what I use — I like the "Board of Education" made by Parallax (**www.parallax.com**). See Figure 2. This board has a small 170 connection point solderless breadboard and an external socket for a BASIC Stamp. It also has a USB port for programming (Parallax also sells a RS-232 version of this board) and a five-volt regulator for providing power to the BASIC Stamp and the electronic circuits.

I have recently obtained the "Professional Development Board" made by Parallax (Figure 3). This board has a large 840 connection point solderless breadboard that has more than enough room for all of my projects. What makes this board exciting to use is the plethora of additional electronic accessories that are included on it. Table 1 shows a list of many of the additional components that come with this board. These are common components that I am always using in my

**Figure 2.** Board of Education with a solderless breadboard from Parallax.

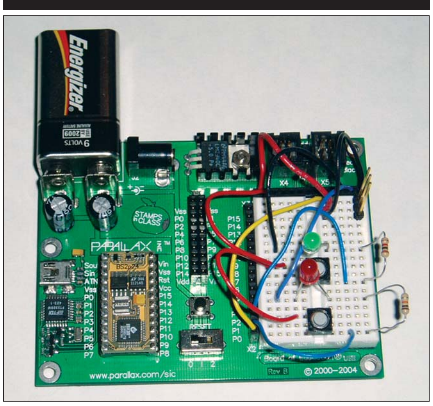
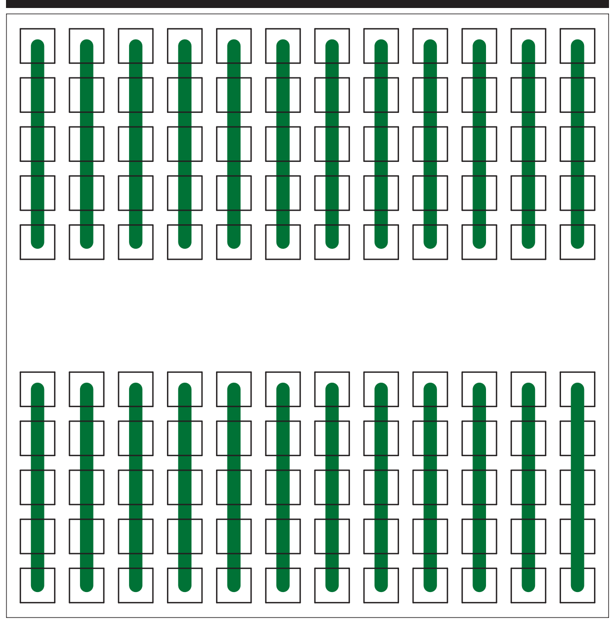



**Figure 1.** Solderless breadboard electrical layout.



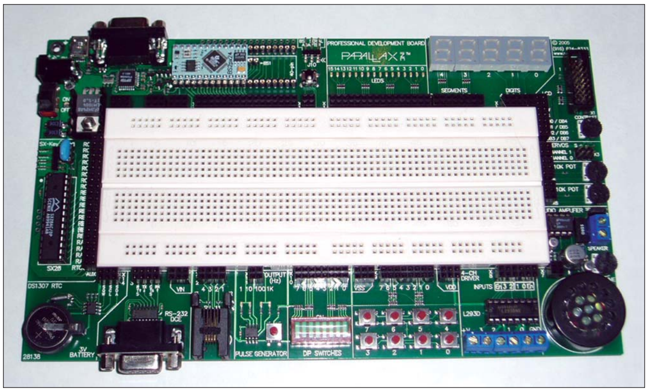


Figure 3. Professional development board from Parallax.

various projects, which are always competing for the breadboard area in my electronic experiments. By moving these common features off the breadboard, there is now more room to do other things. For example, one of my first experiments was programming a SX28AC microcontroller to monitor a quadrature encoder and serially communicate the position to a BASIC Stamp. The built-in sockets for both of these microcontrollers made this a simple wiring exercise. It is amazing that all of this is on a board that measures 5-1/2 inches by 9 inches and only costs $149.

| |
|---|
| 840 Connection point solderless breadboard |
| 14 Pin socket for a BASIC Stamp 1 |
| 24 Pin socket for 24 pin BASIC Stamp compatible microcontrollers |
| 40 Pin socket for the 40 pin BASIC Stamp |
| 28 Pin socket for the SX28 microcontroller |
| 16 Blue LEDs |
| 5 Seven-Segment blue LED display |
| Audio amplifier with speaker |
| Pulse generator |
| 2 10K potentiometers |
| 8 Momentary push buttons with pullup resistors |
| 8 DIP switches with pullup resistors |
| DS1307 Real time clock with battery backup |
| Standard RS232 port for communicating with other devices |
| USB and RS232 ports for programming |
| SX-Key and BS1 programming ports |
| One amp, five volt regulator |
| L293D Motor driver |

Table 1. Parallax professional development board features.

**Q.** I want to build a maze solving robot, but I am having problems trying to get my robot to move the same amount of distance every time. I think the problem might be the servo motors I am using or my batteries. The servos I am using are the TS53 servos I got online from Tower Hobbies, and I am using four AA batteries for power. And the batteries don't seem to last very long. Is there any way I can know for sure that my robot can go the same distance every time?

**— James Patrick**

**A.** The problem that you are having is not related to the type of servos or batteries. When you use these servos for navigating, you tell the servos to turn on for a certain amount of time, then tell them to change directions, and move again for a certain amount of time. For many applications this works fine, but for navigation problems where position and timing are critical, this approach seldom works in a reliable manner.

Three of the biggest sources for the errors are the batteries, surface variations, and wheel slippage. When the batteries start to drain, the voltage drops slightly, and the wheel's speed slows down. In fixed timing routines, unexpected changes in the wheel speed will result in variations in the distance traveled.

Another source of errors is surface variations the robot travels on. For example, robots will lose position accuracy when driving on soft carpets as opposed to hardwood floors. The servos require slightly more torque to drive on the soft pliable carpet surface than on a hardwood floor. The slight increase in torque requirements (especially when turning) will also cause a slight drop in the voltage to the motors, which means they will move slower, again resulting in imprecise distance movements.

Wheel slippage is another problem that all navigation robots must deal with. If the wheel is slipping on the surface, then the robot isn't moving much, which helps to cause the robot to get lost in mazes.

The way people get around these problems is to use encoders mounted on either their motor shafts or the wheels themselves. What the encoders do is provide information on exactly how much the wheel has turned, and depending on how the software is written, it can provide velocity information. Basically, an encoder is a disk with a set of two different colored lines on it, and a sensor is used to count the number of times a color transition occurred across the sensor as the wheel rotates. Figure 4 shows a photograph of a servo wheel with a 32 stripe (16 dark and 16 light) encoder. When the wheel makes one complete rotation, the encoder will detect 32 light-to-dark-to-light color transitions.

When the diameter of the wheel is known, precise distances can be moved just by counting the number of times the encoder detected the color transition, and when the desired encoder count has been reached, the microcontroller tells the servos to do the next task. Timing and battery drain problems are no longer a problem. For example, this servo wheel is 2.61 inches in diameter. Thus, the circumference of the wheel is 8.20 inches. The distance a robot will travel per encoder count is the circumference divided by the number of encoder stripes.

For this wheel, the robot will move 0.256 inches for every encoder count. Now if you wanted the robot to move 24 inches, just divide this number by 0.256 inches/encoder-count, and you get 93.75. Round this number to the nearest integer to get 94 encoder counts. So, all the microcontroller has to do is tell the servo to move forward until the encoder count reaches 94.

Since you are using servos for

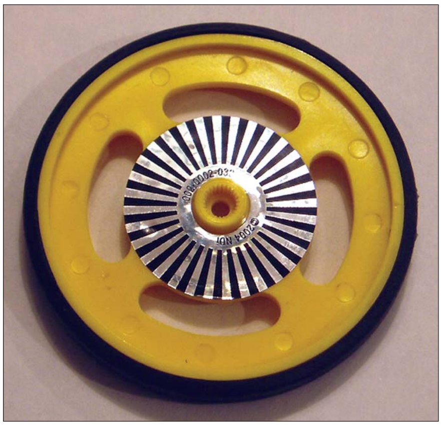


**Figure 4.** The 32 strip encoder disk mounted on a standard R/C servo wheel.

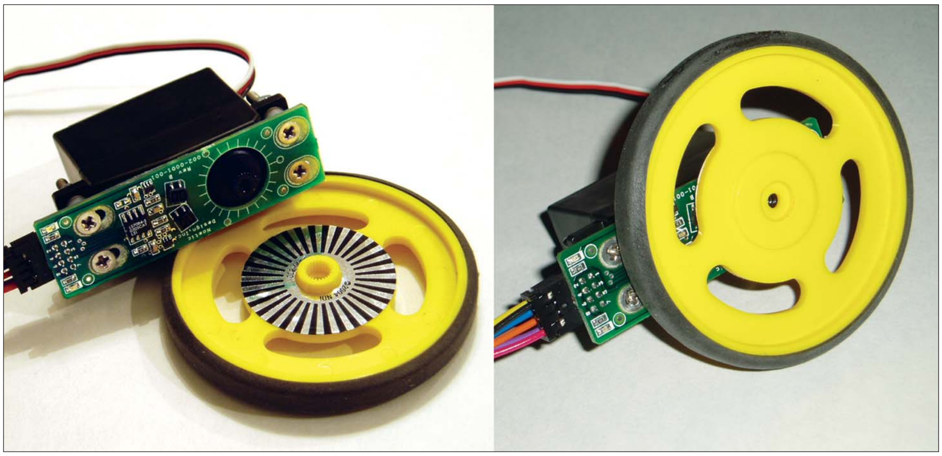


**Figure 5.** WheelWatcher R/C servo encoder system.

your robot, take a look at the WheelWatcher encoders from Nubotics (**www.nubotics.com**), sold by Acroname (**www.acroname.com**). The WheelWatcher is a small pre-fabricated circuit board that mounts directly on your existing servos. It uses a quadrature encoder (two sensors placed 90 degrees out-of-phase) so that rotational direction can be monitored. It comes with a set of spacers so that it can be mounted on a wide variety of servos, and a self sticking encoder disk that is designed to mount directly on the common servo wheels used in the robotics community. Figure 5 shows a photo of this encoder setup mounted directly onto a Tower Hobbies TS-53 servo.

The WheelWatcher was easy to assemble. It took me about 15 minutes to do this, and most of that time was spent looking for one of the spacers I dropped on the floor. A white spacer on a white carpet, a no-win situation. The flat head screws that came with the setup are sufficiently long enough to be able to mount the servo to its structure with no problem. Table 2 lists the wires from this encoder. Not all wires have to be used in your application. I only needed to monitor the ChA and the ChB lines in my test programs.

Interfacing the WheelWatcher to a BASIC Stamp microcontroller was easy (see Figure 6). The program shown in Listing 1 demonstrates that reading and analyzing the encoder measurements doesn't require a lot of programming. The WheelWatcher is probably the easiest way to incorporate encoders with regular R/C servo motors. And using encoders will definitely help you solve your position measuring problems with your robot. More information about how to understand how encoders work and how to use them can be found in *SERVO Magazine* — past and future issues — or you can search the Internet using the key words "robot encoders."

| |
|---|
| Pin 1 — Clock |
| Pin 2 — Vcc (+5V) |
| Pin 3 — Direction |
| Pin 4 — Ground (connects to Pin 8) |
| Pin 5 — ChB (Channel B) |
| Pin 6 — Vcc (Connects to Pin 2) |
| Pin 7 — ChA (Channel A) |
| Pin 8 — Ground |

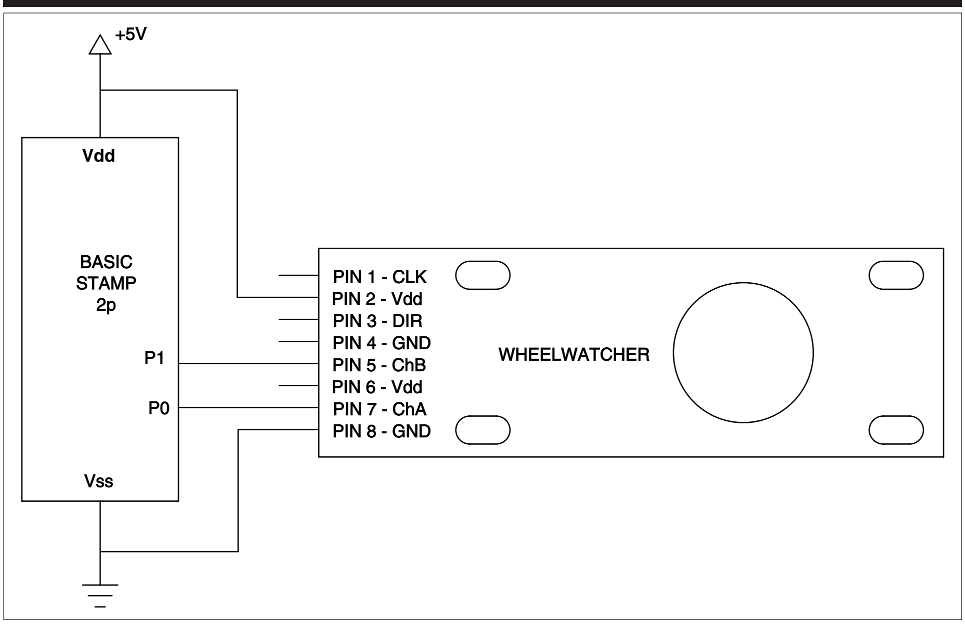
**Table 2.** WheelWatcher signal wires.

**Q.** Are there any tricks to using a soldering iron? Some days I get really good solder joints, and other times it seems to take forever to get the soldering iron hot enough to melt the solder. So, I was wondering if there were any tricks.

**— Gene Alzani**
**Vancouver**

**A.** There really are no tricks to soldering, just make sure that the proper techniques are used and

**Figure 6.** WheelWatcher hooked up to a BASIC Stamp.



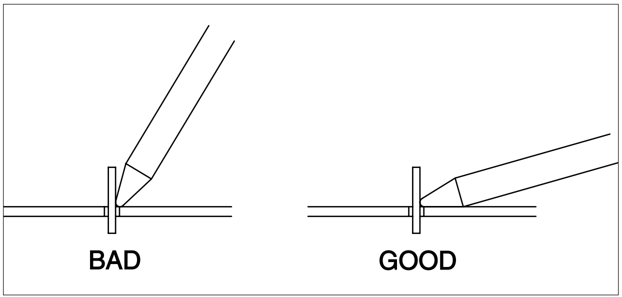



**Figure 7.** Illustration of a good and bad tip placement of a soldering iron.

the surfaces are clean. Dirty surfaces usually lead to poor soldering joints, which eventually lead to joint failure (either mechanical failure or intermittent electrical signal losses). The surfaces can be cleaned with sandpaper, steel wool, non-flammable solvents, clean rags, or an eraser. Any form of corrosion or oxides on the surface should be removed prior to soldering.

Next, you want to make sure that the joints are firmly held in place or attached to each other, such as twisting wires together, bending component leads in circuit boards so they don't fall out, or clamping together. You want to make sure that when the solder is cooling, the components/joints don't move. Movement during the cooling process creates voids and cracks, which eventually lead to corrosion and joint failure.

Part of the cleaning process is to clean the tip of the soldering iron. This is done continuously throughout the soldering process. Any oxides on the tip of the soldering iron make a great insulator, which prevents good heat transfer from the tip to the joint and solder. The tip of the soldering iron needs to be tinned periodically during the soldering operation. After heating up the soldering iron, wipe the tip of the soldering iron across a damp cloth or sponge (damp, not soaking), then add some solder to the tip and wipe the tip clean again. This is known as the tinning process. The tip should look silvery after this is done. Dark or black colored spots remaining on the tip means that there are still oxides on the surface. If it looks crusty, then you may need to turn off the soldering iron and manually clean the crust off the tip.

A good habit to get into is whenever you pick up a soldering iron to solder a joint, just wipe the tip across the damp sponge a couple times then move to the soldering operation. Occasionally re-tinning the tip of the soldering iron helps make cleaner and shinier joints. This only takes about a second or two to do, and it helps keep the iron clean, hot, and ready to use when you need it.

When it comes to soldering, you want to maximize the amount of surface area that is in contact with the soldering iron. Figure 7 illustrates this. The more surface area that is in contact with the surfaces to be soldered, the faster it will heat up and the easier it is to melt the solder. When the surface is hot, apply the solder near the opposite side of the joint/wire relative to the soldering iron and allow the solder to flow towards the soldering iron. If you apply the solder to the tip of the soldering iron during the soldering process, you are more likely to create a poor solder joint, especially if the joint wasn't properly heated. A good solder joint will look smooth and silvery. A poor solder joint will be partly or completely crusty looking.

When working with larger areas or

## Listing 1

```
' Sample program demonstrating how to read and use encoder count
' information from a NuBotics WheelWatcher.  ChA conneced to Pin0
' ChB connected to Pin1.  This program will display encoder count
' and distance moved based on a 2.61 inch diameter servo wheel.

Counter      VAR Word    ' Encoder Counter
prev_state   VAR Byte    ' Previous Encoder Count State
tmp_state    VAR Byte    ' Current Encoder State
Distance     VAR Word    ' Distance Wheel Moved
Conv_Factor  VAR Word    ' Conversion factor to convert Encoder
                         ' counts to distance

Conv_Factor = 8200/64    ' Move distance in micro-inches per
                         ' encoder count.  2.61 inch diameter wheel
                         ' times PI (3.14) times 1000 (micro-inches
                         ' per inch) divided by 64 encoder color
                         ' transitions

prev_state = IN0         ' Get current encoder state
Counter = 0              ' Reset the counter

main:                    ' Main Routing
  tmp_state=IN0                      ' Store current encoder state
                                     '   in a temp variable

  IF tmp_state<>prev_state THEN      ' Check to see if the state changed
    IF IN1 = 0 THEN                  ' If state changed, then make sure
      Counter = Counter + 1          '   that direction has changed to
    ELSE                             '   to know whether the wheel is
      Counter = Counter - 1          '   oscillating, or actually
    ENDIF                            '   moving.
    prev_state=tmp_state             ' Set previous state to current
                                     '   state

    Distance = Counter*Conv_factor   ' Calc moved distance
    DEBUG CLS                        ' Display results in debug window.
    DEBUG "Encoder Counter: ", SDEC Counter, CR
    DEBUG "Distance Traveled: "
    IF Counter.BIT15 = 1 THEN
      DEBUG "-", SDEC -distance/1000, ".", DEC3 -Distance//1000
      ELSE
      DEBUG SDEC distance/1000, ".", DEC3 Distance//1000
    ENDIF
  ENDIF
  GOTO main
```

thicker materials, you may need to use some solder flux. This is a paste that gets brushed on the surface to be soldered, and it helps the solder flow around and through the joint. About the only "trick" I use is putting a tiny amount of solder on the tip of my soldering iron right before I apply the tip to the workpiece. I make sure that this solder is melted before I do this. What this does is when I place the flat surface of the soldering on the workpiece, it helps maximize the contact area between the iron and workpiece, and it causes it to get hotter faster. So, I can solder the joint faster.

**Q.** At the store I see a lot of different types of screw heads from regular flat head screws, to round head screws, to hexagonal screws. I was wondering why there are so many different types of screws.

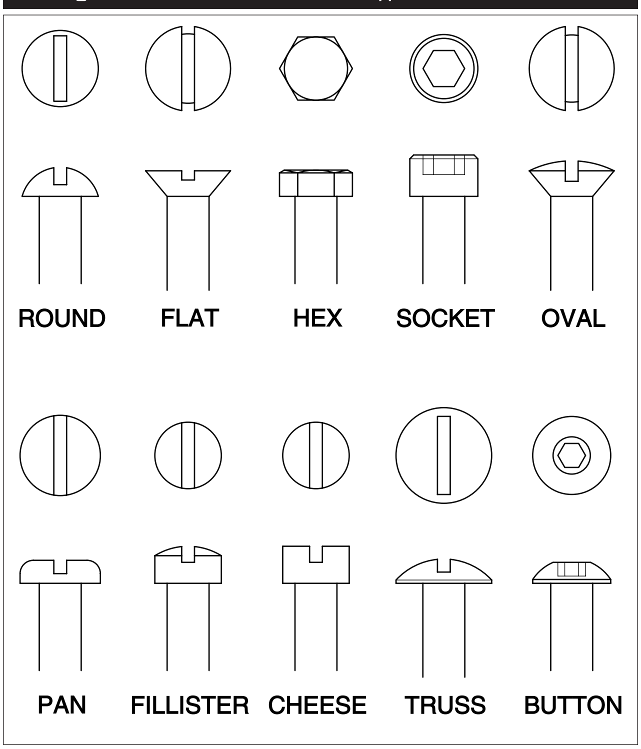
**— Jim Brittan**
**Los Angles, CA**

**A.** It is amazing how many different types of screw heads there are. Round head, flat head, and hex head screws are some of the more popular ones. For the most part, the type of screw that is selected is based more on appearance than function. There are exceptions where space and load carrying capabilities dictate which screw to use. Figure 8 shows a sketch of 10 different types of screw heads all drawn to scale to one another based on a #8-32 screw. A #8 screw has a 0.164 inch diameter, and the –32 means that it has 32 threads per inch.

Flat head, oval head, fillister, and cheese head screws are usually placed in countersunk holes, whereas the other screws are attached to the surface of the holes. Socket head screws (more commonly known as Socket Head Cap Screws, SHCS) are placed either in a countersunk hole or on the top surface of the hole. The countersink depth of the fillister and oval screws are usually to the depth where the oval part of the head protrudes above the surface of the part. Truss screws are usually used for attaching soft or thin materials to another material. The hex, socket, and button head screws are usually used where higher holding forces are needed since wrenches and hexagon sockets can be used to apply greater forces to the screws than with regular screwdrivers.

This is not an all-inclusive list of screw head types, but these are the ones most commonly used and that can be obtained from most hardware stores. For diameters less than 1/4 inch, these screw types can be obtained in brass, steel, stainless steel, and nylon. Larger diameter screws are usually found in steel and stainless steel, and in some cases, nylon. I mention nylon here because it makes a great fastener when you are using expanded PVC for your robot construction materials. They are light, strong, and can be easily shortened/modified with an X-Acto knife. **SV**

**Figure 8.** Various screw head types drawn to scale.



# GEERHEAD

by David Geer

Contact the author at geercom@alltel.net

# The DARPA Grand Challenge 2005

***The DARPA Grand Challenge is a yearly unmanned robotic ground vehicle competition sponsored by DARPA (the Defense Advanced Research Projects Agency). The purpose of the event is to field test robotic ground vehicles in order to advance autonomous vehicle technology.***

"We established the Grand Challenge to help foster the development of autonomous vehicle technology that will someday help save the lives of Americans who are protecting our country on the battlefield," said DARPA director, Dr. Anthony Tether. "The quality of the field that is emerging offers strong evidence that our program is succeeding."

DARPA, a Federal military research agency self-described as "the central research and development organization for the US Department of Defense," held the competition on October 8 and 9 in the Las Vegas, Jean, and Primm areas of southern Nevada.

Competing vehicles were required to travel a 131.6-mile course of rough desert roads with only the aid of onboard sensors and navigation equipment to find and follow the route and avoid obstacles.

Information about the coming event was disseminated gradually, as time grew short and the need to know became apparent.

## The NQE

Before the Grand Challenge Event (GCE), a National Qualification Event (NQE) was held to qualify vehicles. Both the NQE and the GCE finish more vehicles than vehicles finish the race.

Vehicles that competed in the NQE qualifier race had to navigate a two-mile course designed to simulate conditions found on the route for the Grand Challenge Event (GCE). Trials and conditions included "speed runs, tunnels, gates, narrow roads, mountain switchbacks, and man-made and natural obstacles," according to DARPA.

In order to insure an adequate number of competitors, DARPA decided as of July 5 that the 2005 NQE roster would be backed up by nine alternate teams. DARPA used site visits to rank the backup teams. DARPA assessed these vehicles on a standardized test course, separate from and prior to the NQE. The test course was run by these vehicles on August 15.

Three of those teams passed muster and were welcomed into the fold of competitors for the NQE, making a list of 43 finalists.

Finalists were made up of a diverse cross section of individuals and organizations from industry, the R&D community, government, the armed services, academia, high school and college students, backyard inventors, and automotive enthusiasts who were interested in a tough technical challenge and an important national cause, according to DARPA.

These finalists included Axion Racing (Westlake Village, CA), Team Cajunbot (Lafayette, LA), Team CalTech (Pasadena, CA), CIMAR (Gainesville, FL), Team Cornell (Ithaca, NY), Team DAD (Morgan Hill, CA), Desert Buckeyes (Ohio State University, Columbus, OH), Team ENSCO (Springfield, VA), The Golem Group/UCLA (Los Angeles, CA), The Gray Team (Metairie, LA), Insight Ra cing (Cary, NC), Intelligent Vehicle Safety Systems I (Littleton, CO), Mitre Meteorites (McLean, VA), MonsterMoto (Cedar Park, TX), Mojavaton (Grand Junction, CO), Princeton University (Princeton, NJ), Red Team (Carnegie-Mellon University, Pittsburgh, PA),

The Red Team's Sandstorm successfully navigates the NQE qualifying course for the third time.

The "Mojavation" vehicle making its way through the trial course.

"Mojavation" kisses the K-rail and is disabled.

Stanford's "Stanley" heads for a successful completion of the NQE.

Stanley on the course.

Red Team Too (Carnegie-Mellon), SciAutonics/Auburn Engineering, (Thousand Oaks, CA), Stanford Racing Team (Palo Alto, CA), Team Terra Max (Oshkosh, WI), Virginia Tech Team Rocky (Blacksburg, VA), and the Virginia Tech Grand Challenge Team.

In the first round of the NQE, 30 of the 43 Unmanned Vehicles (UMVs) did not finish the race, one didn't pass inspection, and one declined to even try the qualifying run. At the end of the grueling eight-day event, 20 teams and three backup teams were welcomed into the 2005 GCE.

Halfway through the NQE, 22 vehicles had managed to finish the rough terrain and obstacle-laden narrow course at the California Speedway in Fontana, CA — outside Los Angeles — at least one time. You see, the so-called races (the NQE and the GCE) are as much if not more about finishing at all than they are about finishing first.

The NQE ran from Wednesday, September 28, through Wednesday, October 5; the GCE was held October 8 and 9. Performance expectations for the NQE and the GCE were met and surpassed.

"A year ago, I would have been happy to see a robot travel one mile at the NQE," said Tether. "But we have seen a significant number of autonomous ground vehicles traverse a very tough 2.2-to-2.7 mile course more than once — and in some cases, three times!"

"We're going to surprise everybody on October 8 with a fiercely competitive field of worthy vehicles," said Grand Challenge Program Manager Ron Kurjanowicz.

## The GCE

The 23 remaining teams trekked out to the starting location in Primm on October 6 with their robots. October 7, teams were allotted the day to fix any aspects of the vehicles that were broken before the race.

The cash prize for the single best performing vehicle in the GCE was $2 million. Robot vehicles that were allowed to compete had to operate autonomously, navigate the course successfully, and detect and avoid obstacles while moving at "militarily relevant" speeds.

The GCE is a monster of a trial for any vehicle, let alone an experimental unmanned robot. The course covered 150 miles of dirt, uneven terrain, and dried out lakebeds of the Mojave Desert, winding through tight mountain passageways and around manmade obstacles.

Teams were kept from any knowledge of the route the course would take until two hours before the race began at sunrise that Saturday. The 23 teams that competed were selected from 43 semifinalists, who were, in turn, picked from an original field of 195 applicants over a year-long process that required teams to submit technical papers and videos, and to submit themselves to site visits and inspections by DARPA technical experts who would determine the fitness of vehicles for the course.

The race would start and end at the Primm, NV base of operations,

The Cimar vehicle on the course.

Cimar corners against the K-rail and is disabled.

A competition vehicle hugs the wall.


Firefighters douse a hay bail and rescue IRV.

Vehicle heading for a tank trap.


"Ghostrider" is removed from the course.

which was also the finishing ground for the previous year's competition.

DARPA commented prior to the competition that unmanned ground vehicle technology had progressed dramatically over the 18 months before the race and that this progress would readily be demonstrated.

DARPA Director Tether said, "Without question, the teams participating ... this year are as good as or better than the teams fielded in Grand Challenge 2004. All have an excellent chance — we will be held in suspense all day waiting to see whether there will be a winner."

Last year's race — the inaugural competition — took place in March 2004, when 15 teams were selected as finalists to compete for a $1 million prize. The 15 finalists in the 2004 Grand Challenge attempted to autonomously navigate a course between Barstow, CA and Primm, NV. Yet, the competition yielded no winners as no one finished the course. This gave DARPA an opportunity to reserve last year's potential winnings of $1 million for this year while adding another $1 million to make the prize $2 million in all.

To receive the prize, the winning team's vehicle had to successfully complete the route faster than anyone and within a 10-hour period. All vehicles were required to be developed without Government funding.

## Results

There were five finishers this year with the $2 million prize going to the Stanford University team and their UMV, Stanley. Stanley won the $2 million prize because it finished the entire course in the shortest time and under 10 hours. Stanley's finishing time was six hours, 53 minutes, and 58 seconds, according to DARPA.

Two vehicles entered by Carnegie-Mellon University — the Red Team's "Sandstorm" and the Red Team Too's "H1ghlander" finished a close second and third. The Gray Team's "KAT-5" completed the course in seven hours, 30 minutes, and 16 seconds.

The Oshkosh Truck's 16-ton robot vehicle, TerraMax, also finished the course, but on Sunday and with a time over the 10-hour limit.

The top four of the five finishers made history by becoming the first unmanned ground vehicle robots to travel a great distance at relatively high speeds within a specified timeframe. Stanley's average speed over the 131.6-mile desert course was 19.1 mph. Sandstorm averaged 18.6 mph, H1ghlander 18.2 mph, and KAT-5 17.5 mph, according to DARPA; these results prove conclusively that autonomous ground vehicles can travel long distances over difficult terrain at militarily relevant rates of speed.

"These vehicles haven't just achieved world records, they've made history," said Tether. Pointing out that DARPA's mission is to accelerate the development of promising technologies, and then turn them over to others for the development of viable applications, Tether continued, "We have completed our mission here, and look forward to watching these exciting technologies take off." SV

### THAT GRAND CHALLENGE SATURDAY AFTERNOON

As of Saturday afternoon October 8, three vehicles had completed the course and one would receive the prize. The winning team — ultimately determined to be the folks from Stanford University — received a ceremonial check on Sunday for $2 million from Dr. Tony Tether, director of the DARPA Grand Challenge event. DARPA has been authorized by Congress to fund technology competitions like these.

Both the Stanford vehicle, Stanley, and the two Carnegie-Mellon vehicles completed the course on the first day in less than eight hours. Two additional vehicles eventually completed the course.

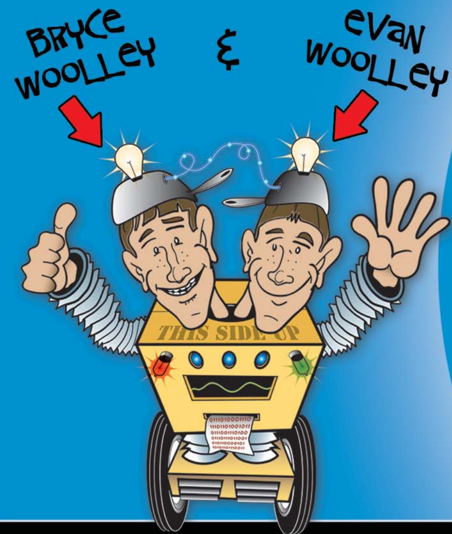
"It's incredible what Stanford and the two Carnegie-Mellon teams did today, and what the other two teams can still achieve," Tether said that Saturday. "We had anticipated from the beginning that we might have to carry the competition over to a second day."

"When the Wright Brothers flew their little plane, they proved it could be done," Tether continued. "And just as aviation 'took off' after those achievements, so will the very exciting and promising robotics technologies displayed here today."

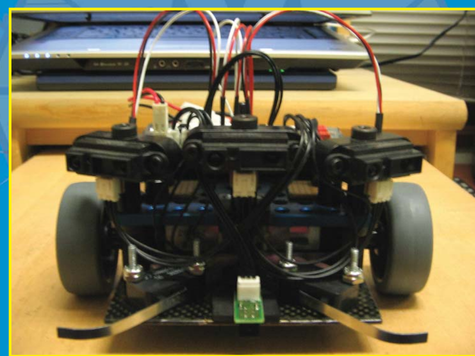
Go to **www.grandchallenge.org/** for more information on the DARPA Challenge.

## Robotics Showcase

# Twin Tweaks




## THIS MONTH:
## Along Came Bobby



The international robotics community — if you thought that we were rambling on about cross cultural collaboration just because we got our hands on two European robot kits a few months ago, think again! We got three! Actually, we were serious about the importance of looking beyond the States when it comes to making progress in robotics, and to prove our commitment in doing just that, we are presenting another article on a European robotics kit.

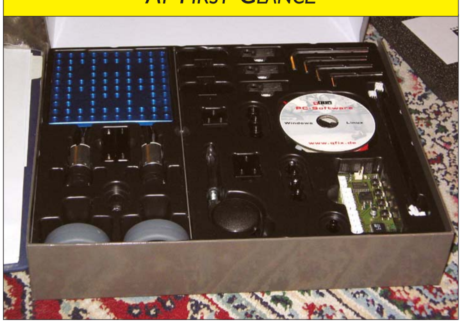
We have the honor of presenting Crash-Bobby, a robotics kit from the German company qfix. Crash-Bobby is a small autonomous robot programmed in C and it comes with software compatible with Windows and Linux operating systems. Bobby also comes with all the parts and tools required for assembly and plenty of room for expansion. Looking at the Bobby box, some hacking might be required for the bot to fulfill some of its "advertised" uses — Bobby Catching, Bobby Clubbing, Bobby Jumping, and Bobby Driving. Some of those suggestions seemed a little weird to us (maybe it is lost in the translation), but Crash-Bobby is indeed a versatile kit.

## Bobby Building

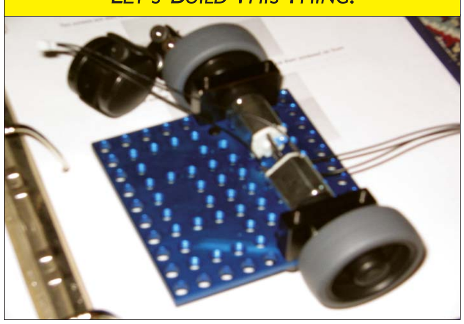
Bobby needs to be assembled, but this is a relatively simple task. The manual provides detailed pictures and clear explanations to help even the most mechanically challenged construct Bobby with ease. And in case you don't feel like practicing your German, all documentation has been blessed with an excellent translation. The simple assembly mainly consists of screwing the prefabricated pieces onto the snazzy blue anodized aluminum base plate.

The kit includes several Allen wrenches to assist in construction and these are really the only tools needed for the construction of the stock kit. The parts, by the way, are of the highest caliber, proving that Crash-Bobby lives up to its moniker of "the professional robotics kit."

AT FIRST GLANCE



LET'S BUILD THIS THING!



## Bobby Crashing

Unfortunately, the stock kit also lives up to its name in other respects — Crash-Bobby. Bobby comes equipped with three Sharp infrared distance measuring sensors for obstacle avoidance, but these sensors are mounted so high on the robot that Bobby has a tendency to trip over or crash into low obstacles.

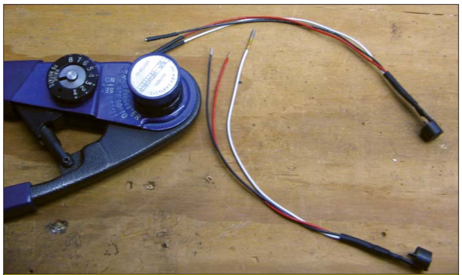
We discovered Bobby's vision problems after downloading the sample program "bobbyTest." Downloading programs to Crash-Bobby is done through a parallel port and the somewhat unintuitive tool Pony Prog 2000. Pony Prog is also labeled appropriately — it greets the user with a startling whinny every time the program is launched. The sound can be turned off, but that is the only action that can be executed to make Pony Prog easier to use. Setting up this downloading tool is not covered in detail in the instruction manual, but a clever tinkerer should have little trouble calibrating the tool properly. Once the tool is set up, downloading programs is an easy task.

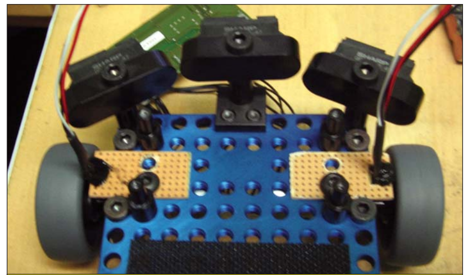
An interesting side note is that the binary form of the program is the one actually downloaded to Bobby. We looked at the binary code in the program editor one time and it freaked us out. Total gibberish (it was even more difficult for us to read than the German) — C syntax seems so much nicer now.

The bobbyTest program provides a simple obstacle avoidance behavior using the Sharp distance measuring sensors. While it's entertaining for a little while to block the little robot with your hands and make it move in the other direction, it made us kind-of feel like bullies. Bobby was simply too easy to trip up — low obstacles were not being picked up by the Sharp sensors, and most of the time the low obstacles were still big enough so that Bobby could not traverse them. To give Crash-Bobby fancier footwork and more of a chance to avoid the social gaff of tripping in public, we decided to equip the "professional robotics kit" with some professional grade sensors and materials.

## Bobby Sensing

Of course, an easy solution to Bobby's problem would be to mount the Sharp sensors closer to the bottom of the robot, but it's our prerogative to hack the robot, not just rearrange it! The next easiest solution would be to add some kind of sensor to the bottom of the robot. The kit we received came with a line following sensor, but that might lead to trouble if Bobby picks up on a line that leads to a dead end. To make Bobby a more discriminating trailblazer, we also decided to add some limit switches, but merely bestowing the sense of touch didn't seem to do justice to Bobby after such a long trip. The final touch would be two Hall-Effect sensors that would act as encoders for the wheels. For the sake of specificity, we decided to use ATS660LSB True Zero Speed, Hall-Effect Adaptive Gear-Tooth Sensors.


HALL-EFFECT SENSORS.


SENSOR PLACEMENT.

To keep with the professional caliber of the rest of the kit, we decided to mount our sensors on a custom carbon fiber base plate. Carbon fiber is an interesting material to work with — instead of cutting it with a jigsaw, we opted for the finesse of aviation shears (caution kids: don't try this at home unsupervised; carbon fiber splinters are not fun). Some careful work with a drill press resulted in a refined sensor panel that was worthy of attachment to the professional robotics kit.

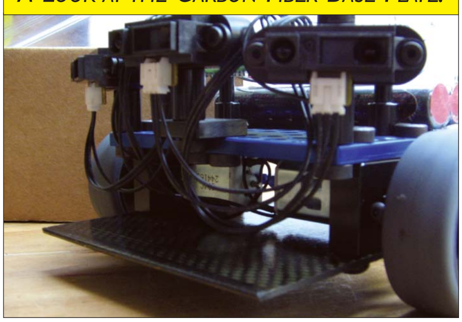
One of the great things about Crash-Bobby is that the body of the robot has been designed to make structural expansion a very painless operation. The base plate comes choc full of .200" holes (metric — it is a European robot, after all), some that are smooth and others that are already threaded. The motor mounts also have threaded holes top and bottom, and we picked up the bottom holes for the attachment of our custom carbon fiber sensor panel.

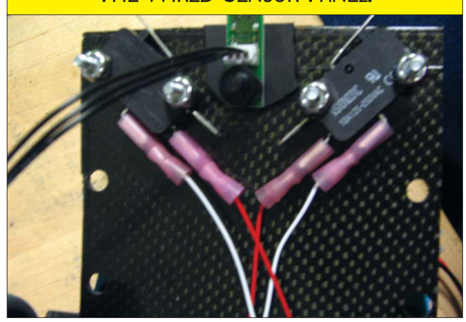
The Hall-Effect sensors were fixed to the robot via mounting pieces cut from perforated board. Our only lapse in usage of professional grade materials was with the type of silicone that we used for adhering the Hall-Effect sensors to the perf board — it was a black substance not usually used for electronic parts due to its somewhat corrosive properties. We utilized the utmost caution and we were able to redeem ourselves with the rest of the neat wiring.

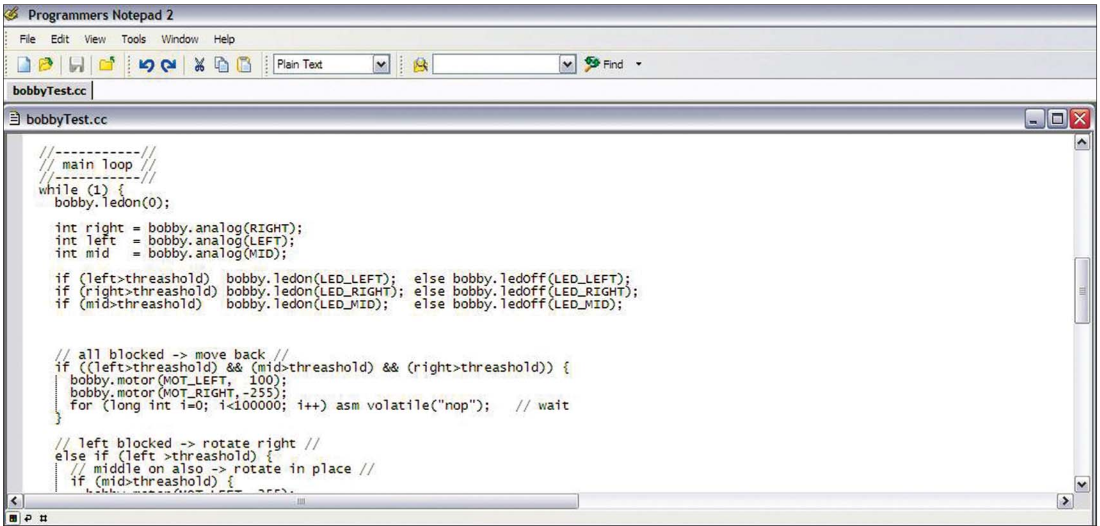
Instead of using any old connectors for our hacked-on sensors, we chose high quality Mil-spec connectors so as to avoid impugning Crash-Bobby's professional reputation. Mil-spec connectors demand a Mil-spec crimper, and Bobby's additional wiring was done in a way so as to reflect the quality of the entire kit. For those that like technical details, the Mil-spec sockets we used were M39029/57-354 22 AWG sockets. We didn't bind the wire sockets together with any sort of plastic casing or anything — the sockets alone provided a solid connection to the ports on Bobby's circuit board.

By adding the line following sensor, the two limit switches, and the pair of Hall-Effect sensors, we made use of all of the free ports on Bobby's circuit board. The board comes with two motor ports, four analog ports, and four digital ports. All are inputs, which silenced an early idea for a music-making Bobby that would have produced tunes via a toy organ circuit that received input from the Sharp sensors. The creative process is an interesting thing.

We plugged the line following sensor into the last analog port and wired

A LOOK AT THE CARBON FIBER BASE PLATE.


THE WIRED SENSOR PANEL.




Programmers Notepad 2

File Edit View Tools Window Help

bobbyTest.cc

```
//-----------//
// main loop //
//-----------//
while (1) {
  bobby.ledOn(0);

  int right = bobby.analog(RIGHT);
  int left  = bobby.analog(LEFT);
  int mid   = bobby.analog(MID);

  if (left>threashold)  bobby.ledOn(LED_LEFT);  else bobby.ledOff(LED_LEFT);
  if (right>threashold) bobby.ledOn(LED_RIGHT); else bobby.ledOff(LED_RIGHT);
  if (mid>threashold)   bobby.ledOn(LED_MID);   else bobby.ledOff(LED_MID);


  // all blocked -> move back //
  if ((left>threashold) && (mid>threashold) && (right>threashold)) {
    bobby.motor(MOT_LEFT,  100);
    bobby.motor(MOT_RIGHT,-255);
    for (long int i=0; i<100000; i++) asm volatile("nop");   // wait
  }

  // left blocked -> rotate right //
  else if (left >threashold) {
    // middle on also -> rotate in place //
    if (mid>threashold) {
```

Some Bobby Code.

nity presented itself for Bobby to make a public appearance. Chaparral High School's robotics club CREATE (Chaparral Robotic Engineers and Techno Explorers) had an opportunity to gain some exposure by setting up a booth at the local mall and showcasing their projects. Yeah, we're college students now, but the robots keep bringing us back. They have a magnetic quality.

At the booth, we displayed five different robots, and Crash-Bobby was among those presented as our latest robotics project. One interesting distinction that set Bobby apart from our other creations was that it was the only one to embody the strictest definition of a robot. According to Webster's dictionary, a robot is "any mechanical device that performs complex, often humanlike actions automatically or by remote control." The dictionary actually offers the looser (but still very valid) definition of a robot. The strictest definition maintains that a robot should be autonomous.

Crash-Bobby was the only completely autonomous robot at our booth, and it arguably demonstrated the simple yet somewhat humanlike behavior of obstacle avoidance. Bobby's slow and steady pace made it look like quite the cautious traveler, and its abrupt changes of direction when it sensed an obstacle seemed sort-of spontaneous. It was a big hit with kids that had never seen an autonomous robot before. Crash-Bobby's appearance at the mall seemed to scare up some interest in the kit as well, so perhaps Bobby has a bright future in the States. He certainly made a good impression, and we were proud to display the hacked Bobby as a project representative of our robotics club.

## Bobby Programming

Neither of us are great programmers (hopefully college and good old practical experience will change that), but thankfully that isn't a prerequisite to mess with

the limit switches and Hall-Effect sensors into the digital inputs. In an ideal situation, we may have wanted the line follower also on a digital port, but we decided that it was more important for the other sensors. The limit switches, which have only the two states of on or off, are certainly not fit for an analog interpretation and our scheme of how to turn the Hall-Effect sensors into encoders demanded a digital approach.

Crash-Bobby's wheels are attached to the motor shafts via a set screw, and by replacing the screw that came in the kit with a longer one, we created a surface that modeled the tooth of a gear for the Hall-Effect sensors to sense. The theory behind our encoders was simple: the sensors would read high upon seeing the screw head and read low when not receiving input.

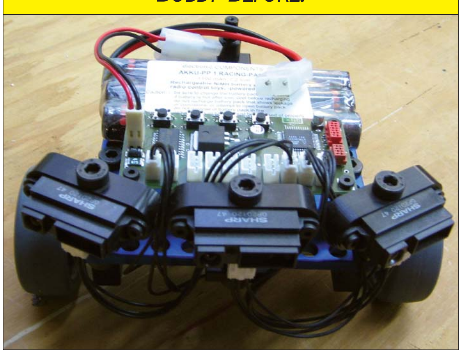
The Hall-Effect sensor is normally used for sensing the profile of a gear, and it essentially does this by counting teeth. The sensor measures the magnetic gradient created by the passing of a ferrous object, usually a gear tooth. The sensor converts this measurement to a voltage for digital reading. We used a fancy sensor that sought to minimize the effects of the air gap (space between the tooth and the sensor) and temperature drift (variations in temperature), so we should get good readings.

Accurate navigation is dependent on accurate information, and we couldn't have the professional robotics kit going astray because of some bad sensory data. Of course, that would be great if we could figure out how to program the Hall-Effect sensors properly, but even if we're not up to the challenge we're sure that a tinkerer out there has some great ideas about how to get it to work.

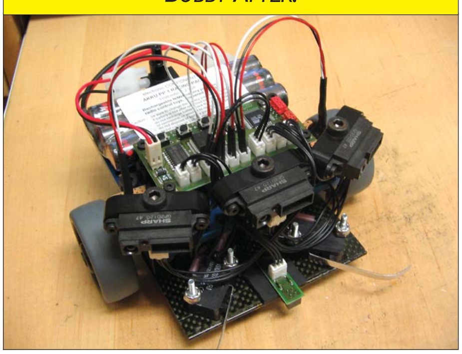
## Bobby Showing

Before we had a chance to program a more sophisticated obstacle avoidance behavior into Crash-Bobby, an opportu-

Bobby Before.



Bobby After.



Bobby's user-friendly code. The code has been meticulously commented so even programming novices know what's going on. All we had to do to get the limit switches to work was define a few new variables and copy and paste some chunks of code. In theory, anyway.

The Programmers Notebook is also a fairly simple Integrated Development Environment, and the Bobby kit comes with a wide variety of sample programs that should help tinkerers gain a quick mastery of the commands unique to the kit. The instruction manual also comes with a short tutorial on programming and a glossary of command terms that is handy for quick reference.

Programming certainly demands a degree of trial and error (a crash-course, if you will) when first starting with Crash-Bobby, but remember that this is coming from novice programmers. We are confident that Crash-Bobby is a programming platform that is suitable for both beginning and veteran programmers. In our opinion, the excitement of seeing a robot move around based on your code is a much more interactive way to learn programming than to watch computer applications run themselves — it gives quite a sense of tangible accomplishment to see a robot act the way you want it to.

If anything, working with Crash-Bobby should motivate novice programmers to hone their programming skills — the spry robot is undeniably fun to work with.

## Bobby Testing

Testing a new program is also a trial and error experience. We have a pretty good handle on C syntax, so that even when what would be considered an ineffective program was compiled, it was built smoothly with no errors. A smoothly compiled program doesn't necessarily mean a smoothly running program, however, and Bobby exhibited some interestingly erratic behavior as we tried to gift it with the tactile sense.

Sometimes Bobby would move backwards despite any sensory input from the switches or the Sharp sensors. Other times, Bobby would only react to sensory input by jerking backwards. Bobby's madness did not stop there — sometimes it would simply move around in such unpredictable directions that it left us scratching our heads and looking for some calculus homework to give us a break from the tedium of programming.

After many cycles of inspiration, frustration, and said tedium, we finally had a working program for the hacked on limit switches. It was about time, too — there is quite a large obstacle to space ratio in our dorm room, if you know what we mean.

But what about the Hall-Effect sensors? They could have a variety of applications. For example, the sensors could be used to grace Bobby with an effective dead reckoning program. The sophisticated Hall-Effect sensors have the ability to sense the direction and frequency of rotation for the wheels. The frequency measurement could quickly be translated into a gauge of speed and distance via simple equations that take into account the circumference of Bobby's wheels and the speed of the motors. Knowing values like distance and speed would lay the foundation for a dead reckoning program that could give Bobby direction in lieu of sensory input.

As far as obstacle avoidance, these abilities might not be particularly useful, but the Hall-Effect sensors could certainly be put towards more refined navigation techniques — ones that don't require bumping into everything in sight (or out of it).

We also equipped Bobby with a line following sensor, but we didn't really have enough time to do much with it. Besides, line detection has been done before, and we're not ones to follow the crowd. The Hall-Effect sensors are so much more interesting, anyways.

## Bobby Marching On

In short, Crash-Bobby is an excellent kit for those intrepid tinkerers looking to hack or expand a robot structurally, electronically, and/or programmably. As a fun and interactive way to learn programming, a platform for sensory experimentation, or simply as a hobby project, Crash-Bobby has a lot to offer American roboticists. In the end, we have determined that Crash-Bobby does indeed deserve the title of "the professional robotics kit," and we think that it should be readily embraced by professionals and hobbyists alike looking for a high quality kit to tinker with.

Finally, we would like to extend a special thanks to Dr. Stefan Enderle of qfix for the honor of working on Crash-Bobby and as a gracious representative of the European robotics community. SV

*For more information go to:* **www.qfix.de**

Build Your Own

# Planetary Rover Bogey

## Suspension Platform

**by Alan Federman**

**B**oth the Mars Pathfinder and MER rovers used a rocker bogey suspension system of six wheels. The rocker bogey is an amazingly simple design that has proven its effectiveness. It is passive — no computer is required to control it. The goal of the system is to keep as many wheels in contact with the ground as possible over uneven terrain. I wanted an easy-to-build, inexpensive platform for students to use to simulate planetary rover operations.

My requirements were the platform had to be able to carry a payload of up to 10 kg, the entire robot could be lifted by one person, and that it could fit in the average car trunk. My target price for the platform would be $200-500, depending on how good a scrounger or recycler I turned out to be.

I've seen lots of student projects where someone has put a camera on a RC car, and called it a rover, but I don't feel that teaches kids anything about robotics. Most commercial kits are only suitable for use on smooth level surfaces indoors. Now, where's the fun in that? If I am going to build a rover, I want one that can get its wheels dirty! I call my robot the PMMP = Pneumatically Mediated Mobile Platform. I am going to show you how I built it.

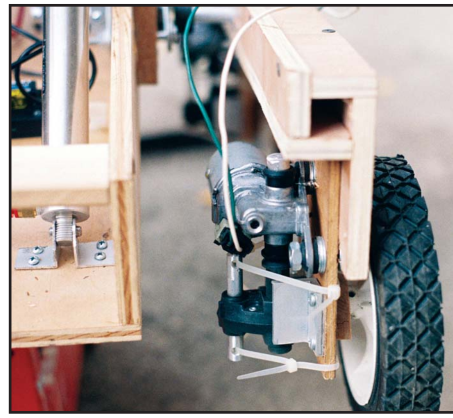
### Why Go Bogey?

Just think about it. Suppose you had a four wheel vehicle with no suspension. If one wheel hit a rock or a curb and tried to climb it, you'd soon have two wheels up in the air, and only 50% of your traction. If your suspension could automatically adjust to the terrain, you'd always have at least three wheels on the ground and 75% or more of your traction. A rule of thumb is a round wheel can't climb a curb generally greater than 1/3 of its diameter. A four wheel system with a bogey suspension can usually get by at a 1/2 to 2/3. Add a rocker to the bogey, and you can usually climb an obstacle 1 to 1.5 diameters. Another advantage of a bogey suspension is the platform tends to stay level with the ground. So a flush mounted camera is always aligned with the local horizon.

There are several ways to make a bogey. NASA's rovers use differential gears. You could also use simple tie rods, like the control rods used by RC modelers. For my bogey, I chose pneumatic cylinders, because I could get away with being a sloppy machinist, and I happened to have a couple of extra ones lying about. New cylinders are kind-of expensive, but used ones aren't. If you were really strapped, you probably could make your own by using a pair of screen door closers or bicycle pumps.

### Chassis

I am a big fan of using plywood box beams for structural elements. Plywood is cheap, easy to work with, and almost everyone can get access to a table saw. I



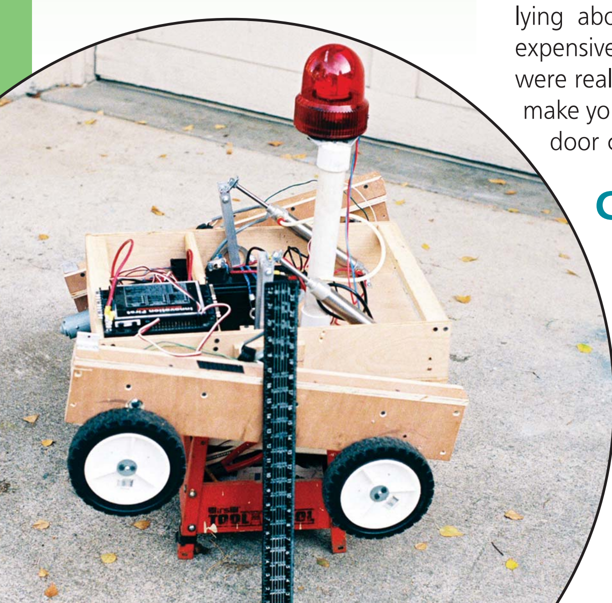
**This photo shows the end of the box beam, made of three 1.5 inch wide pieces of 1/2 inch birch plywood, glued and screwed to one piece of six inch wide plywood. The motors are bolted to the wider piece of plywood. Two assemblies — one for each side — are made.**

started by making two 30 inch box beams to hold the motors. I used four 12 volt seat positioning motors. These use a two stage worm gear reduction for a top speed of about 75 RPM. I used lawnmower wheels mounted directly to the output shaft by a pin and endcap tapped into the shaft end.

The Central chassis is just an open plywood box. A few pieces of lumber are used to stiffen the box and act as support struts for the battery and warning light.

Both wheel assemblies are attached to the central chassis box by ball bearing races and 8" aluminum shafts. The shafts are either drilled through and bolted, or drilled and tapped to attach additional structural elements.

Before attaching the pneumatic cylinders, it is important to check the balance and freedom of motion of the wheel assemblies. The cylinder attachment brackets are made from some L channel, some C channel, and some 1/2: aluminum rod. The assemblies are either bolted directly to the plywood box or drilled, tapped and bolted together. I used a small piece of plumber's steel



**Complete view of the rover showing the freedom of motion and the legs at opposing angles.**

## PARTS LIST

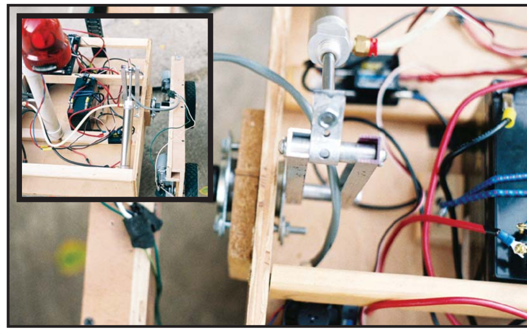
- 1 4' by 4' sheet of 1/2" birch plywood
- 36" long 1/2" diameter aluminum shaft
- 2' of 3/4" aluminum C channel
- 1' of 2" aluminum L channel
- 1 box 1" drywall screws
- 1 bottle carpenters glue
- 12" of Plumber's steel mounting tape
- Two pneumatic cylinders, with tubing and fittings
- 2 1/2" inside diameter self-locking ball bearings, with mounting flange.
- Four 6" diameter lawnmower wheels
- Four matched low speed 12 volt automotive motors (seat positioning or similar)
- One 12 volt rechargeable battery (sealed lead acid is best)
- Small fuse panel and fuses
- Two PWM ESCs
- Wires and crimp connectors
- RC or other controller

**Tools**

- Table saw
- Good cordless electric drill, with screw driver attachments
- Hacksaw
- Files and sandpaper
- Drills, including a good sharp 1/2" bit
- Crimp tool, wire cutter/stripper

**Optional Tools**

- Drill press
- Small tap set
- Dremmel
- Soldering iron



Detail of bearing and cylinder attachment.

mounting tape to attach the end of the cylinder to the top of the axle lever assembly.

## Final Assembly

Hook-up the two cylinders straight across (top to top, bottom to bottom). Check to see that the operation is what you expect. When one side tilts up, the other should tilt down. I suspect that using pneumatics instead of a differential may have some advantages, especially if you are driving a little more quickly than is prudent by NASA standards. The mechanism is sloppy and works like shock absorbers for sharp bumps. I am using a 12 volt battery (15 amp hour capacity) typically sold in electronic stores and used in small uninterruptable power supplies. I am using two older Innovation First Victor 800 series PWM speed controllers, but this is probably overkill. A pair of smaller hobby shop ESCs should work just fine.

I have run this platform with both an older Innovation First FRC controller and a RadioShack VEX.

## Future Plans

Now that I have my platform, I intend to try some enhancements, including different autonomous controllers, a solar panel battery charger, and maybe a stereo camera system. So now I have my very own planetary rover, and I am all set to explore my back yard. You can have your very own, as well. I hope my design will inspire your creativity. I say a computer on every desk and a robot in every garage! **SV**

# Programmable Logic Part1

by Gerard Fonte

The goal of this new bimonthly series is to provide a basic understanding of the various programmable logic techniques.

There are a lot of powerful low-cost components available today that are rarely considered by hobbyists — and even some engineers — because of unfamiliarity.

You have to be comfortable with the idea and concepts of programmable logic before you will be likely to employ them.

We'll start with the simple applications, such as diode logic, and work our way up to ASIC (Application Specific Integrated Circuits). We'll also discuss multiplexer logic, PROM (Programmable Read Only Memory) logic, PROM arithmetic, PALs (Programmable Array Logic), PLDs (Programmable Logic Devices), and CPLDs (Complex Programmable Logic Devices). In addition, we will examine various tools that are useful.

This includes basic concepts — such as Sum of Products — as well as advanced ASIC software. This month is diode logic, discrete transistor logic, and multiplexer logic.

## Basic Logic Concepts

Fundamentally, digital logic consists of controlling a signal that has two discrete values. Most often, a logic "0" (or logic low) is a level that is about zero volts. A logic "1" (or logic high) is typically around five volts. The actual voltages depend upon the application. There is always a range between logic levels that is disallowed. This separation between the levels is necessary to avoid confusing one state (logic level) for the other.

It is very important to point out that there is a very big difference between a logic "0" (that is at zero volts) and an open circuit, which also measures zero volts with your voltmeter. The difference is that a logic "0" is pulled to ground and will sink current. An open circuit cannot sink (or source) current and is at a high impedance state. This third state is neither "0" nor "1." It is usually called "Z," "Hi-Z," "Open," or "Tri-state." We will see later that this state has its uses, too.

## Diode Logic

The simplest form of diode logic is the "OR" circuit that is shown in Figure 1. It's easy to see that if any input is high, the output will be high, as well.

FIGURE 1: An OR gate using diodes. Any input driven high makes the output high.

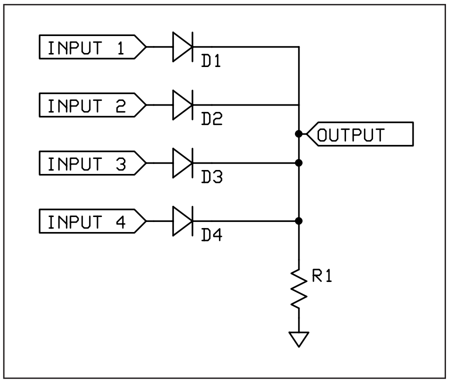
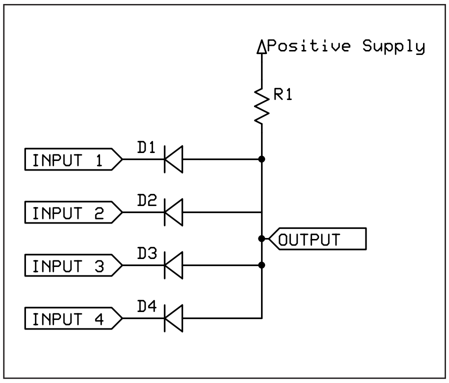



FIGURE 2: An AND gate using diodes. When all the inputs are high, the output is high.



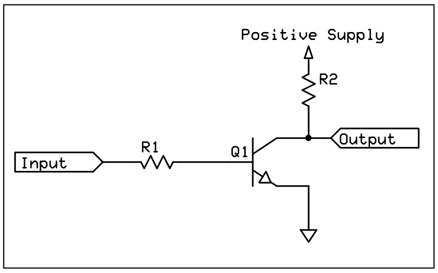



FIGURE 3: A simple inverter. The output is the opposite of the input.

This is also called the "Wired OR" circuit. Note that R1 plays the very important role of pulling down the output if all the inputs are low. Otherwise, the output would be in a Hi-Z state. Additionally, the resistor speeds up the response time. Diodes can have a considerable capacitance. If all the inputs are low, any positive voltage at the output will have no place to go. The only way for the charge to dissipate is through the leakage of the diode junction. This can be 100s of megohms. It's obviously much faster to discharge this through a resistor to ground.

The other simple form of diode logic is the "AND" circuit that is shown in Figure 2. It's clear that every input must be high in order for the output to be high. This is called the "Wired AND" circuit. In this case, R1 has a current limiting function. It is also assumed that the inputs will be true logic values and not Hi-Z. If an input is a Hi-Z, then an input pull-up resistor (to the positive supply) will be needed to speed up the response.

AND and OR circuits are okay, but limited. It would be really nice if we could have a NAND or NOR circuit, as well. But, in order to do this we need an inverter, which you can't do with just diodes. For that, you will have to use the circuit shown in Figure 3. This is a simple transistor inverter. When the input goes high, it turns on the transistor and it pulls the output low. If you connect the input of this circuit to the output of the AND or OR circuits, you will get NAND or NOR functions, respectively.

Once you have NAND or NOR logic, you have the ability to create any and all logic functions. This includes flip-flops, counters, memories, etc. From a practical standpoint, this has limited value. If you need a complex counter, it's faster and cheaper to buy one. However, if you need a high-powered flip-flop for your robot, making one out of discrete diodes and power transistors may make sense.

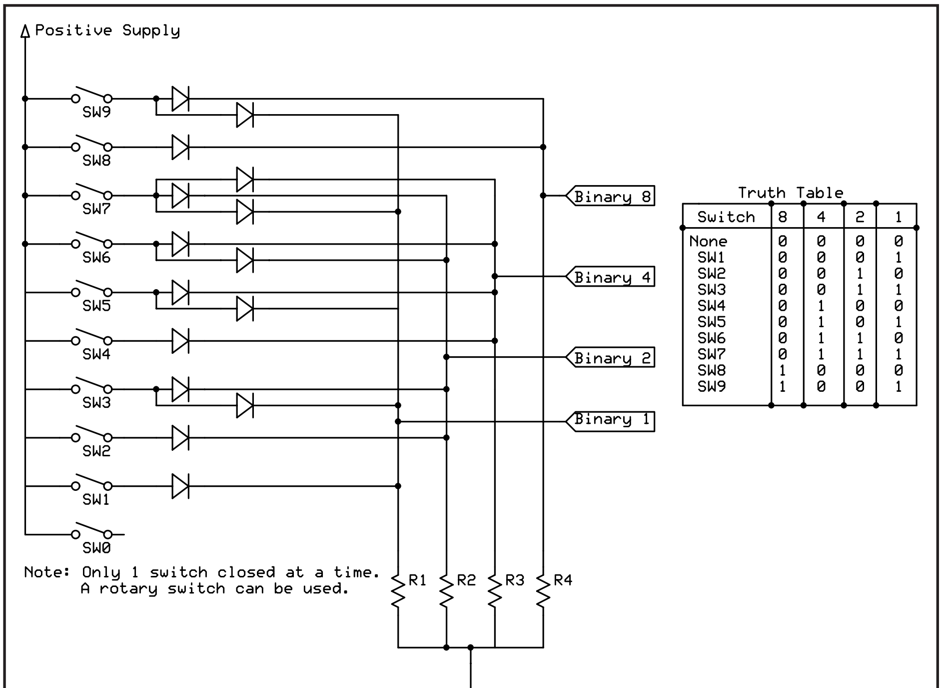



FIGURE 4

FIGURE 4: A diode matrix converts a decimal value into a binary value.

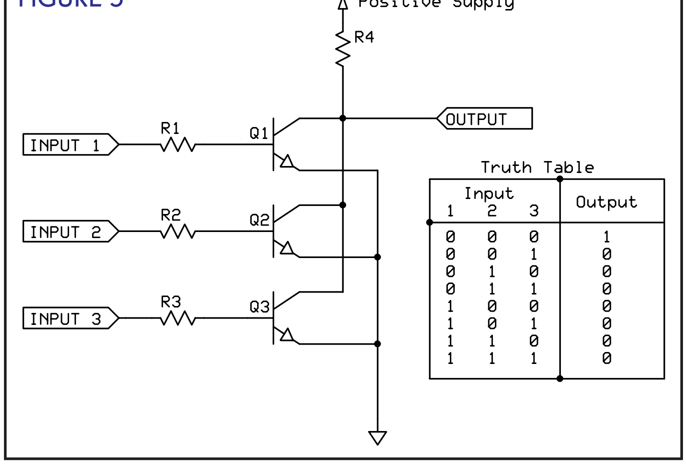



FIGURE 5

FIGURE 5: The NOR function as implemented with discrete transistors.

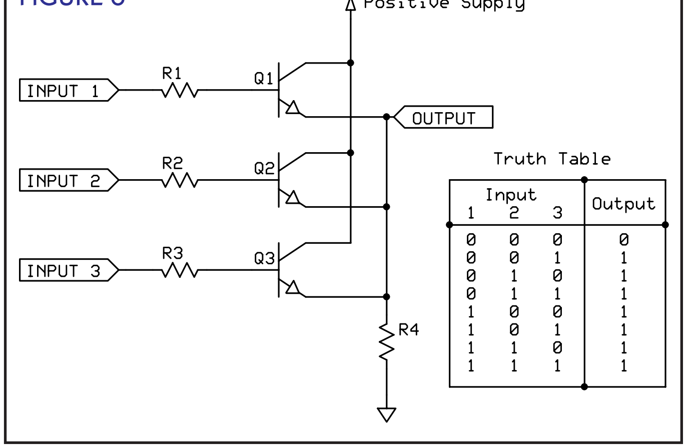
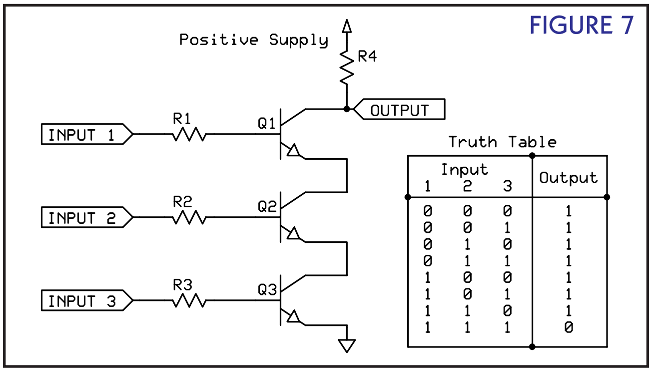



FIGURE 6

FIGURE 6: The OR function using discrete transistors.

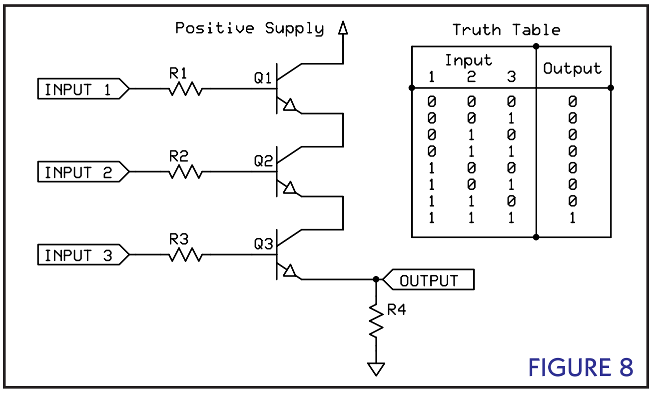
## Diode Matrix

An important and useful type of diode logic is called a matrix. This is an array of diodes that typically convert one of many inputs into a binary value. Figure 4 shows an example of a diode matrix that converts a decimal value to a binary value. This is very useful for keyboards and rotary switches. Note that only one



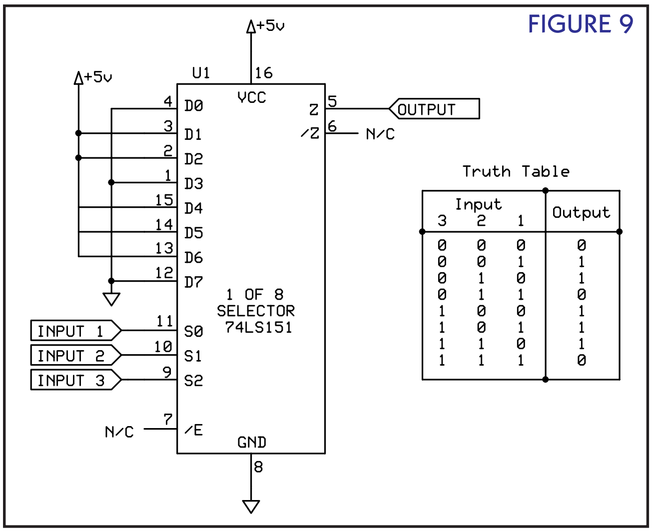

| Input 1 | Input 2 | Input 3 | Output |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

FIGURE 7: Using transistors to make the NAND function.



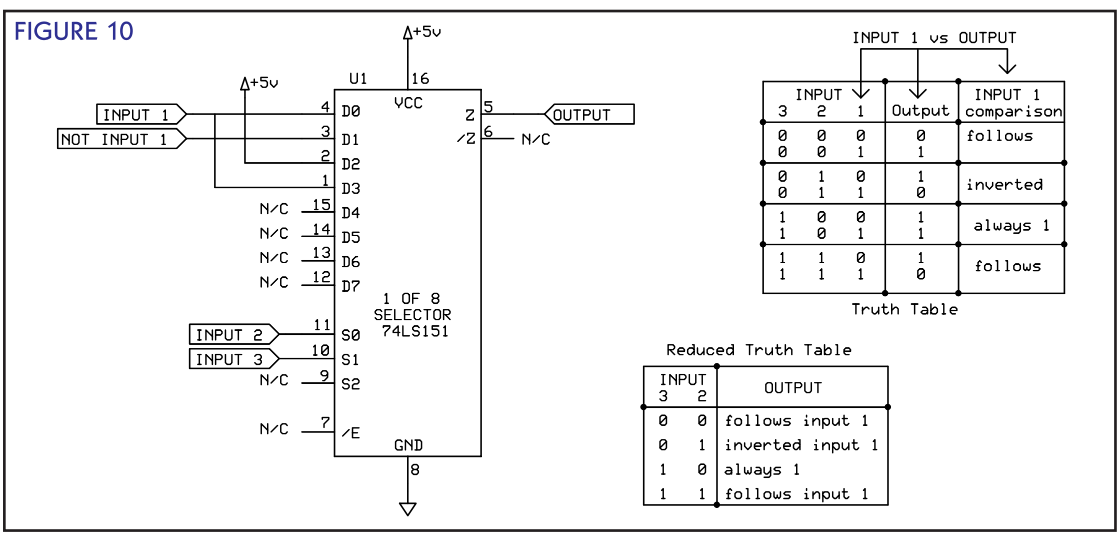

| Input 1 | Input 2 | Input 3 | Output |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

FIGURE 8: Using transistors to make the AND function.




| Input 3 | Input 2 | Input 1 | Output |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

FIGURE 9: A multiplexer can be used to create any logic function.




| INPUT 3 | INPUT 2 | INPUT 1 | Output | INPUT 1 comparison |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | follows |
| 0 | 0 | 1 | 1 | |
| 0 | 1 | 0 | 1 | inverted |
| 0 | 1 | 1 | 0 | |
| 1 | 0 | 0 | 1 | always 1 |
| 1 | 0 | 1 | 1 | |
| 1 | 1 | 0 | 1 | follows |
| 1 | 1 | 1 | 0 | |

Reduced Truth Table

| INPUT 3 | INPUT 2 | OUTPUT |
|---|---|---|
| 0 | 0 | follows input 1 |
| 0 | 1 | inverted input 1 |
| 1 | 0 | always 1 |
| 1 | 1 | follows input 1 |

FIGURE 10: A non-obvious technique lets a two-input multiplexer create a three-variable logic function.

switch may be closed at a time for proper operation.

While Figure 4 may initially look complicated, its operation is straightforward. We can see that the matrix is really just a number of OR circuits connected together. For example, the binary value "2" is high if switch 2, 3, 6, or 7 is closed. SW-0 is shown for clarity because with any OR circuit, if there is no input that's high, the output is low. Of course, you don't have to convert the input to binary. You can use a gray-code (where only a single bit changes at a time) or make up your own code to suit your needs. That's the power of programmable logic. You make precisely what you need.

## Transistor Logic

Transistors can be used in place of diodes for the "Wired OR" circuit. However, since NPN transistors are faster and easier to fabricate than PNP types, there is usually an inversion in the logic.

If you examine Figure 5 you will see that any positive input pulls R4 low. In this case, we have the NOR function.

The employment of transistors provides for amplification, which can be a very useful feature. Additionally, there is better isolation from the input signal because the input resistors can be much greater than those used for diode logic.

Note that the circuit can sink much more current than it can source. The source current is limited by R4. The sink current is limited by the characteristics of the transistor (and heatsink) used.

By moving R4 and the output point (as shown in Figure 6), an OR function is created. This circuit allows you to source much more current than you can sink.

You can fabricate NAND and AND functions by using transistors in series, as shown in Figures 7 and 8, respectively. But you will be limited in how long the series can be because each transistor will reduce the voltage by some degree. This voltage drop depends upon the load and type of transistor. Note that it is possible to use FETs (Field-Effect Transistors) instead of bipolar types. FETs act much more like low-value resistors and can have much less voltage drop. Inexpensive FETs are available that can handle 100s of amps.

## Multiplexer Logic

Multiplexer logic is probably the most simple programmable logic that provides for arbitrary truth tables. You can easily create any transfer function you want. Figure 9 shows how this works. Conceptually, it's very easy. Your three logic inputs are connected to the "SELECT" inputs of the 74LS151 1-of-8 selector (or multiplexer). These select pins determine which "DATA" input is transferred to the output. These data inputs are fixed at either logic high or logic low. In this way, a high or low is transferred to the output, depending on how the data inputs are wired. In the case of the 74LS151, any logic (or Boolean) function of three variables can be easily created. The 74LS150 is a 1-of-16 selector, so any function of four variables can be made with this.

There is a method of increasing the variable size per input. That is, it is possible to use a 1-of-4 selector (instead of 1-of-8) to create any function of three variables. The only additional requirement is that one of the select inputs must be inverted. The procedure is fairly subtle and not immediately obvious. It is shown in Figure 10.

(*Note that I only used one-half of the 74LS151 for clarity. In actual practice, a 1-of-4 selector like the 74LS153 would be chosen.*)

The idea is to break up the truth table into pairs that isolate one variable (typically the LSB or Least Significant Bit). This is done with the horizontal lines in the Figure 10 truth table. Then compare these output pairs to the selected variable. There can be only one of four possible outcomes. Either the pair follows the variable, is the inversion of the variable, is always "1," or always "0." As you can see, all four outcomes are not always present. (In some cases, the inversion may not be required.) These pairs represent the LSB of the select function (or INPUT 1 in the figure). Therefore, the truth table can be reduced to only four entries instead of eight (see Figure 10 again). Once the table is reduced, it is clear how to wire the circuit. The remaining two variable — INPUTs 2 and 3 — go to the appropriate SELECT pins and the DATA input pins are connected as shown in the reduced truth table.

This is a very useful and important technique. It reduces the number of input lines which halves the number of internal logic elements or "gates." We will see in later articles how the reduction of gates increases speed and decreases cost.

## Summary

This month, we just got our feet wet with a few examples of do-it-yourself logic. Hopefully, you found something interesting or new. Next time, we'll be touching on PROM logic and PROM arithmetic along with an introduction to PALs. Eventually, we'll be designing full-blown ASICs. SV


# SINGLE BOARD COMPUTERS: PC/104 FORM FACTOR

| MANUFACTURER | Model | Speed (MHz) | Processor | DRAM/SD RAM, Max, Mbyte | Socket for Bootable Disk OnChip, SRAM, or EPROM | Compact Flash Support | 10/100 Ethernet Controller | RS-232 Serial Ports | RS-422/485 Support | USB Ports | LPT Port |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ***AIC-IES*** www.aicies.com | IB104+ | 800 | TM5800 Crusoe | 64 | Yes | No | Yes | 2 | No | 1 | Bi-Directional |
| ***Diamond Systems Corporation*** www.diamondsystems.com | Prometheus | 100 | 486-DX2 | 32 | No | Yes | Yes | 4 | No | 2 | ECP |
| ***Kontron*** www.kontron.com | MOPS/386A | 40 | 80386SX | 4 | Yes | No | Yes | 2 | No | 0 | Bi-Directional |
| | MOPS/686+ | 266 | Intel Pentium MMX | 256 | Yes | No | Yes | 2 | No | 1 | SPP/EPP/ECP |
| ***Micro/Sys*** www.embeddedsys.com | SBC1670 | 520 | Intel PXA270 | 128 | No | Yes | Yes | 5 | Yes | 1 | No |
| | SBC1495 | 133 | STPC Atlas | 64 | Yes | Yes | No | 2 | Yes | 1 | Bi-Directional |
| | SBC1390 | 25 | 386EX | 16 | Yes | Yes | No | 4 | Yes | 0 | No |
| ***Octagon Systems, Corp.*** www.octagonsystems.com | 2040 | 40 | ALI 386SX | 4 | Yes | No | Yes | 2 | No | 0 | Bi-Directional/ EPP/ECP |
| | 2050 | 128 | 5x86 | 32 | No | Yes | No | 2 | Yes | 0 | Bi-Directional |
| | 2060 | 300 | AMD Geode GX1 | 256 | No | Yes | Yes | 2 | Yes | 2 | Bi-Directional/ EPP/ECP |
| ***RTD Embedded Technologies*** www.rtd.com | CMi6486DXHR | 60 | 486DX | 32 | Yes | No | No | 2 | Yes | 0 | Bi-Directional/ ECP |
| | CMM26686HX300HR | 300 | Geode MMX | 256 | Yes | No | No | 2 | Yes | 0 | Bi-Directional/ EPP/ECP |
| | CME46786HX400ER | 400 | Celeron | 512 | Yes | No | Yes | 2 | Yes | 2 | Bi-Directional/ EPP/ECP |
| ***VersaLogic Corporation*** www.versalogic.com | Jaguar | 566 | Celeron | 256 | Yes | No | Yes | 2 | Yes | 2 | Bi-Directional/ EPP/ECP |
| | Lynx | 133 | AMD SC520 | 54 | No | Yes | Yes | 2 | Yes | 0 | Bi-Directional |
| ***WinSystems, Inc.*** www.winsystems.com | PCM-SC520 | 133 | AMD SC520 | 256 | Yes | Yes | Yes | 4 | Yes | 0 | EPP/ECP |
| | PPM-TX | 266 | Pentium MMX | 256 | Yes | No | Yes | 4 | No | 1 | EPP/ECP |
| | PCM-SX | 33 | 386SX | 8 | Yes | Yes | No | 2 | Yes | 0 | EPP/ECP |
| ***Zendex*** www.zendex.com | ZXE-ST586/104P | 133 | ST Micro Atlas 586 | 128 | Yes | Yes | Yes | 2 | Yes | 2 | ECP |

*Note: The PC/104 Form Factor has board sizes that are 3.6 inches by 3.8 inches. All of these single board computers (SBCs) have their own BIOS and will run DOS. Most will run the Win 98/NT/CE operating systems, along with Linux.*

# MATRIX

by Pete Miles

Upcoming topics include SBCs and H-bridges, sensors, kits, and actuators. If you're a manufacturer of one of these items, please send your product information to: **BrainMatrix@servomagazine.com** Disclaimer: Pete Miles and the publishers strive to present the most accurate data possible in this comparison chart. Neither is responsible for errors or omissions. In the spirit of this information reference, we encourage readers to check with manufacturers for the latest product specs and pricing before proceeding with a design. In addition, readers should not interpret the printing order as any form of preference; products may be listed randomly or alphabetically by either company or product name.

| Floppy Disk Support | Keyboard Controller | Mouse Port | IDE Interface Controller | EIDE Interface Controller | 16 bit PC/104 Expansion Bus | 32 bit PC/104 Plus Expansion Bus | PCMIA Connector | Video | Watchdog Timer | Real Time Clock | Counter/Timers | Digital I/O | Analog Input | Analog Output | Speaker Port | Power |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Yes | AT | No | Yes | No | Yes | Yes | No | SVGA | Yes | No | No | No | No | No | No | 5VDC |
| Yes | PS/2 | PS/2 | Yes | No | Yes | No | No | No | Yes | Yes | 2 | 24 | 16 | 4 | No | 5VDC |
| Yes | AT | No | Yes | No | Yes | No | No | LCD | Yes | No | No | No | No | No | No | 5VDC |
| Yes | AT | No | No | Yes | Yes | Yes | No | SVGA | Yes | Yes | No | No | No | No | No | 5VDC |
| No | 6 pin Keypad | No | No | No | Yes | No | No | LCD | Yes | No | No | 6 | No | No | Yes | 5VDC |
| No | PS/2 | PS/2 | Yes | No | Yes | No | No | SVGA | Yes | Yes | No | 6 | No | No | Yes | 5VDC |
| No | AT | No | No | No | Yes | No | Yes | No | Yes | Yes | 3 | 5 | No | No | No | 5VDC |
| Yes | AT | No | No | No | Yes | No | No | No | Yes | Yes | 3 | No | No | No | Yes | 5VDC |
| Yes | AT | AT | No | Yes | Yes | No | No | No | Yes | No | No | No | No | No | Yes | 5VDC |
| Yes | PS/2 | PS/2 | No | Yes | Yes | No | No | SVGA | Yes | Yes | No | 24 | No | No | Yes | 5VDC |
| Yes | AT | No | Yes | No | Yes | No | No | No | Yes | Yes | 3 | No | No | No | Yes | +5, ±12VDC |
| Yes | AT | PS/2 | No | Yes | Yes | No | No | SVGA | Yes | Yes | 3 | 18 | No | No | Yes | +5, ±12VDC |
| Yes | AT | PS/2 | No | Yes | Yes | No | No | AGP-SVGA | Yes | Yes | 3 | 18 | No | No | Yes | +5, ±12VDC |
| Yes | AT | PS/2 | Yes | No | Yes | Yes | No | AGP-SVGA | Yes | No | No | No | No | No | No | 5VDC |
| Yes | AT | PS/2 | Yes | No | Yes | Yes | No | No | Yes | No | 2 | No | No | No | No | 5VDC |
| Yes | AT | PS/2 | Yes | Yes | Yes | No | No | No | Yes | Yes | 3 | No | No | No | Yes | 5VDC |
| Yes | AT | PS/2 | Yes | Yes | Yes | Yes | No | No | Yes | Yes | 3 | No | No | No | No | 5VDC |
| Yes | AT | No | Yes | No | Yes | No | No | No | Yes | Yes | 3 | No | No | No | Yes | 5VDC |
| No | PS/2 | PS/2 | Yes | No | Yes | Yes | No | SVGA | Yes | Yes | No | 16 | No | No | No | 3.3 & 5 VDC |

*Note: Stacking auxiliary modules will add additional capabilities that are listed here, as well as some that are not. The manufacturers shown offer several other SBCs.*

# NEW PRODUCTS

## CIRCUIT BOARDS

### Lead-Free PCBs

Advanced Circuits — the country's leading source for quick-turn printed circuit boards — has announced the availability of lead-free solder as a plating finish. This is good news for those customers requiring a value-priced alternative to other, more costly lead-free finishes. The new lead-free solder finish provides an enhanced solderable finish advantageous for assembly and meets all legislation requirements for the European Union's RoHS Directive deadline of July 1, 2006. As an added bonus, Advanced Circuits is offering their lead-free finish without a price increase.

"As an innovative and customer-centric company, we focus on determining designer and industry needs well before they become a pressing issue," said Tony Garramone, corporate training manager at Advanced Circuits. "We strive to stay ahead of the curve and provide our customers with the most current technology. We've accomplished this and are excited to offer our customers lead-free options far before requirements mandate."

The European Union's RoHS directive standards were created because the use of lead in electronics represents both an environmental and human health concern. Even though these standards were originated in Europe, the US is gradually following the lead and companies both supplying and building electronics in Europe must be compliant. Right now there are only a few US shops, such as Advanced Circuits, that are currently offering a lead-free solder finish.

Advanced Circuits' lead-free boards are produced using a lead-free alloy of 99.3 percent tin/0.6 percent copper with a trace of nickel (SN100CL). This solder offers an excellent alternative to more expensive lead-free finishes and provides a flatter pad surface than its leaded counterpart. Advanced Circuits is also stocking a variety of higher temp laminates for use in conjunction with the lead-free solder finishes which are designed to withstand processing temperatures between 260° and 288° C, depending on the laminate system the designer chooses. The new lead-free solder is also compatible with other leaded solder components.

Advanced Circuits plans to run its current tin lead solder process in tandem with a lead-free process until the use of leaded solder becomes untenable.

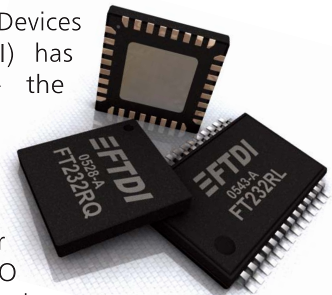
Advanced Circuits has no minimum order requirements and specializes in expedited services including same-day and weekend turns.

For further information, please contact:

**Advanced Circuits**
Tel: **800•979•4PCB**
Website: **www.4pcb.com**

## ELECTRONICS

### FT232R — USB UART IC




Future Technology Devices International Ltd. (FTDI) has unveiled the FT232R — the next generation of their popular USB-UART Bridge family. This highly integrated device includes onboard EEPROM, master clock generator, 3.3V LDO regulator, reset generator, and USB termination resistors. Only two external decoupling capacitors are required for a minimal configuration. In addition to the full set of modem control signals, the device features five I/O pins which can be configured in EEPROM to have several functions including providing a output clock which can be used to drive an external MCU or FPGA. Each device has a unique number (the FTDIChip-ID™) burnt into it at manufacturing time which cannot be altered by the end user. Though an encryption scheme, this feature can be used by product designers to protect their application software from being copied. FT232R features a wide range of royalty-free FTDI developed drivers for 32 and 64 bit operating systems including Windows, CE, Linux, and MAC-OS.

FT232R comes in both standard SSOP-28 and miniature QFN-32 5mm x 5mm package options and is competitively priced at US $1.80 in 10K quantities. A version of the device with a parallel FIFO interface (p/n FT245R) is also available. Please go to the company website listed below or email them for further details on this product.

For further information, please contact:

**Future Technology Devices International Ltd.**
USA Branch Office
5285 NE Elam Young Pkwy., Ste. B800
Hillsboro, OR 97124
Tel: **503•547•0988** Fax: **503•547•0987**
Email: **us.sales@ftdichip.com**
Website: **www.ftdichip.com**

## ROBOT KITS

### Knight Invader III

The future of toys has arrived. Knight Invader III is a super race car that includes mini-construction projects, teaches the basic principles of electricity, gear mechanisms, and demonstrates three different modes of propulsion.

This sleek super racer has it all and a little bit more. KNIGHT INVADER III can blow a fierce force of air to propel it off the starting line and challenges any contender to try to stay up when it is changed into direct motor drive. When on a mission, KNIGHT INVADER III can secretly change its transportation mode to solar power (sold separately). Plus, for the educationally minded guardians, this fire plug kit includes mini construction projects that teach the basic principles of electricity including experiments involving series and parallel circuits and solar energy (sold separately). Another brilliant idea by Dr. OWIKIT is the inclusion of a bulb and switch that allows friends to investigate simple experiments in conductivity. No tools required for assembly. Suggested selling price is $15.95.

For further information, please contact:

**Robotikits Direct**
17141 Kingsview Ave.
Carson, CA 90746
Tel: **310•515•6800** Fax: **310•515•0927**
Email: **info@owirobot.com**
Website: **www.owirobot.com**

## SENSORS

### PicoSonic Ranger

The PicoSonic Ranger-1 (PSR-1) is an ultrasonic, atmospheric time-domain reflectometer, usually referred to as an ultrasonic SONAR sensor. The PSR-1 uses a single ultrasonic transducer element as both an ultrasonic projector and an ultrasonic microphone. Active transducer damping allows detection of objects as close as one foot or less, or as far away as 20 feet to 1" resolution.

There are two basic operating modes that can be selected under hardware or software control. "Normal" mode is an enhanced form of echo ranging that digitally enhances echo contrast and reduces clutter. The "motion detect" mode recognizes and records new target echo range while masking recurring echo signatures. This capability can be used to qualify targets, or "clear the field" of observation of superfluous or redundant information that would otherwise burden the host system.

Measurement data is presented in two formats: 1) as a "framed," real-time echogram suitable for timing by the host; and 2) as an ordered string of distance measurements in ASCII format sent over its asynchronous serial interface.

In addition to mode selection, the user can also program transmit pulse length, receiver gain profile, and receiver mute time delay, which is retained in non-volatile memory for recall on power up based on hardware select states. The PSR-1 consumes <20mw at five volts.

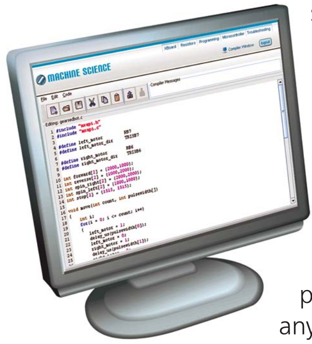
For further information, please contact:

**Eutectos Incorporated**
4250 Deer Meadow Ln.
Occidental, CA 95465
Tel: **707•874•2003**
Email: **info@eutectos.com**
Website: **www.eutectos.com**

## SOFTWARE

### Online C Development System

Machine Science, Inc. — a non-profit organization in Cambridge, MA — has released a new on-line




system for completing microcontroller-based projects — everything from beginning circuit design to advanced programming. This first-of-its-kind system enables users to write and compile microcontroller code using a simple on-line interface and then program their chips without any locally installed software.

"This is a breakthrough technology for hobbyists and educators," according to Machine Science Executive Director Sam Christy. "With our unique on-line system, users can easily get started writing C code for their projects without the usual hassles associated with locally installed IDEs, and compilers." Users simply log in to the Machine Science website and write their code in a full-featured text window. They can then compile their code and send the compiled hex files to a microcontroller with a single mouse click.

There are numerous advantages to using an on-line development environment for microcontroller projects. All code files are stored on a server, which can be accessed from any Internet-connected computer. Machine Science also provides useful code libraries for driving external devices, such as liquid crystal displays. Although the system currently works only for PC users, Machine Science is developing support for Mac and Linux users, as well.

As an additional resource for system users, Machine Science has developed a series of detailed project guides, featuring step-by-step fabrication and programming instructions. Five projects are available, including a wireless text messenger, a music synthesizer, and a programmable robot. More new projects are coming soon. According to Christy, the on-line system currently supports the PIC16F877, but future releases will support more PICs and the full line of AVR processors.

For further information, please contact:

**Machine Science**
Tel: **617 • 354 • 7171**
Website:
**www.machinescience.org**



## Show Us What You've Got!

Is your product innovative, less expensive, more functional, or just plain cool? If you have a new product that you would like us to run in our *New Products* section, please email a short description (300-500 words) and a photo of your product to:

**newproducts@servomagazine.com**

# Stepping Motor BASICS

by Richard Panosh

*Stepping motors are very common in everyday products ranging from wristwatches, printers, disc drives, gas pumps, and machine tools — just to mention a few. Hobbyists are increasingly using them in robotics projects as they become more readily available and their unique capabilities are required. Unlike conventional motors, they can provide precise velocity profiles and position control with open loop circuitry. Stepping motors can also be used in closed loop applications were high acceleration is required and variable loads are involved.*

*Stepping motors are a special class of electric motor without a commutator. Generally, all windings are part of the stator (the stationary part of the motor) and the rotor is either a permanent magnet or a block of magnetic material with teeth, in the case of a variable reluctance motor. Thus, the motors are mechanically simple, but a somewhat complex electronic driver is required to pulse the magnetic field in the proper sequence to achieve motion.*

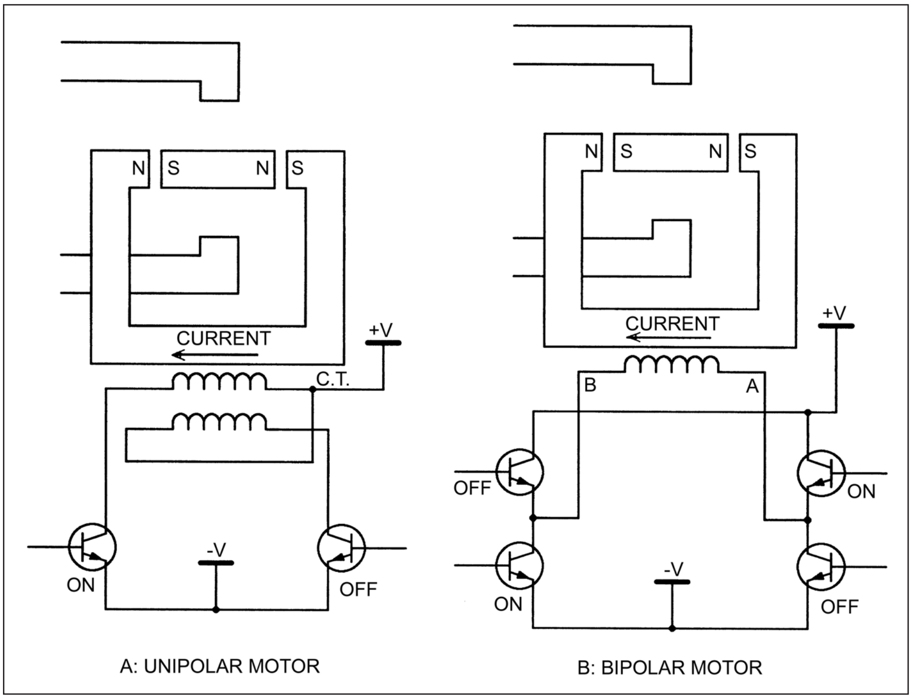



**Figure 1.** Unipolar Motor Winding and Bipolar Motor Winding.

## Introduction

Because stepping motors have matured over a long period of time, there are many variations and options to consider. Indeed, it is very difficult to remember all of them. Already I have described the differences between the permanent magnetic and variable reluctance rotors. In addition, these two types of rotors can be combined into a third hybrid type. In this type, the rotor is axially magnetized with two sections: one containing all the north poles and the other the south poles. You can identify this type of motor by turning the unpowered rotor by hand. Since it is made of soft iron, no magnetic resistance will be felt. But in the permanent magnet type, a definite detent can be felt as the rotor is turned (termed "cogging") which can present problems in some applications. The hybrid and variable reluctance rotor are the most common type.

Two processes can produce the magnetic field of the stator coils. They are referred to as the unipolar and bipolar motor. To begin with, a stepping motor moves one step when the magnetic field of the stator changes direction. The direction of the magnetic field is established by the direction of the current flow in the coil. In the case of the unipolar motor, the current in a stator coil is defined in only one direction so that two coils are required to reverse the field. In the case of a bipolar motor, the direction of the field is simply changed by reversing the direction of the current through a single coil. The operations of these two types are illustrated in Figure 1.

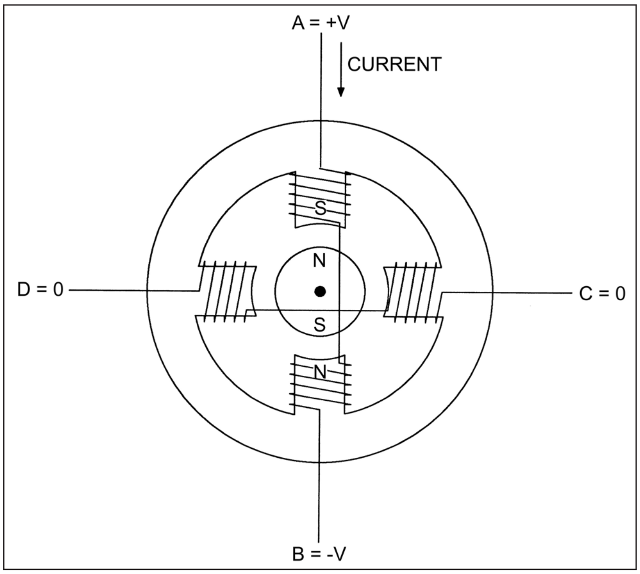



**Figure 2.** Simplified Bipolar Permanent Magnet Stepper.

## Talk of Torque

For a motor to be useful, it must produce torque. The torque of the stepping motor is proportional to the strength of the magnetic field produced by the stator coils. In turn, the magnetic field is proportional to the number of turns in the winding and the current flowing through the winding. This quantity is often referred to as the ampere turns product. Generally, the current flowing through the coil resistance is the limiting factor, as this power loss will begin to heat the motor. This illustrates the advantage of the bipolar motor over the unipolar. For the same coil density, the bipolar stator wire can be made $\sqrt{2}$ larger. This will reduce the resistance of the coil by the inverse amount and permit 40% greater current. Since the torque is proportional to the current, a bipolar motor will therefore produce 40% greater torque.

Figure 2 illustrates a much simplified bipolar permanent magnet stepping motor. Up to this point, we have only illustrated a single direction for the stator coil. Thus, the rotor could only flip back and forth through 180°. The motor in Figure 2 can provide rotation in either direction with a step size of 90°. A stepping motor like this would not provide very precise control and would not be very useful. Real stepping motors have multi-toothed rotors that resemble a gear to provide as little as 48 steps per revolution, to as many as 400. A typical motor may provide 200 steps per revolution, or 1.8° per step.

Mechanical manufacturing methods, design, and tolerances affect the absolute accuracy of a stepping motor. Absolute accuracy depends upon accu-

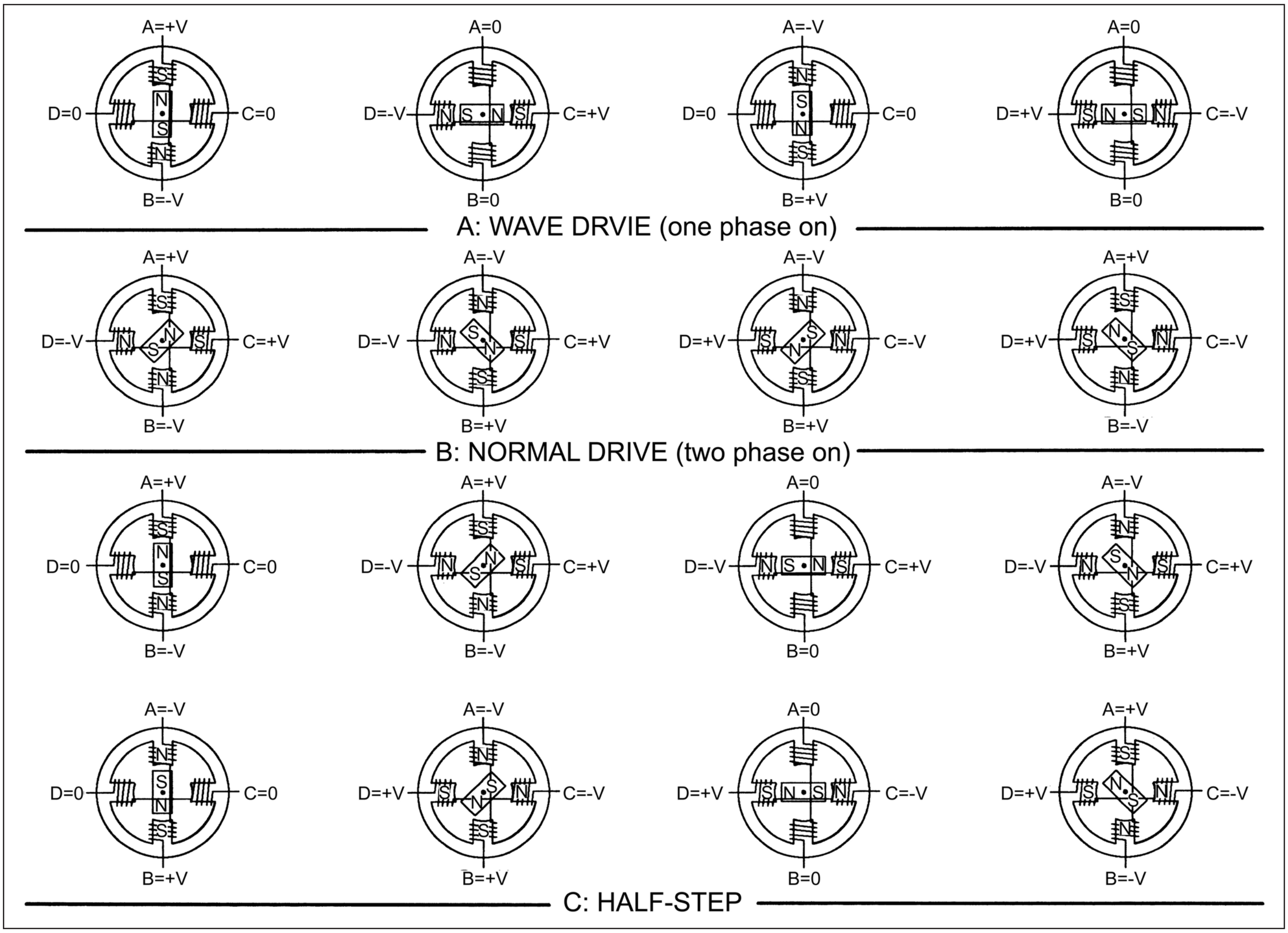



**Figure 3.** Three Possible Drive Sequences.

rate stator teeth location, uniform rotor teeth location, and uniform air gaps. Manufacturing tolerances allow typically ±5% step accuracy. On a step size of 1.8°, this results in an error of ±0.09°.

The motor illustrated in Figure 2 allows three possible current pulse sequences. Each of these sequences is illustrated in Figure 3. Reading left to right, the pulse sequence produces clockwise rotation of the rotor. Rotation in the counter-clock wise direction can be obtained by reversing the order of the pulse sequence.

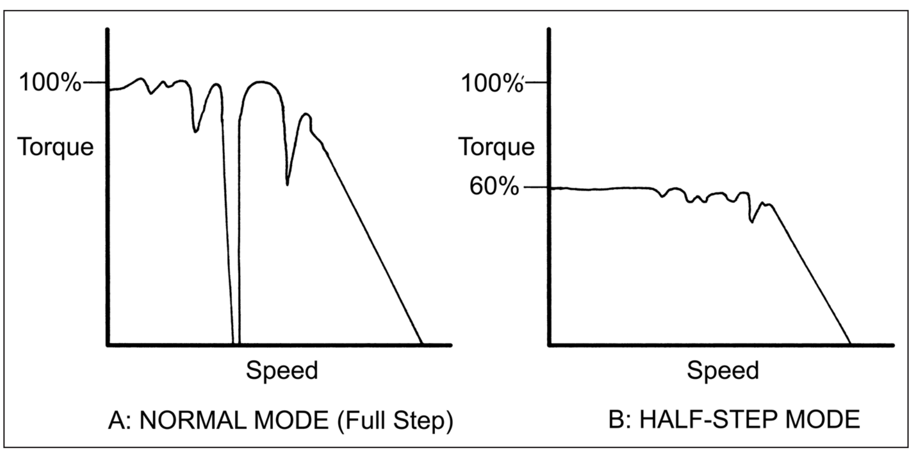
Figure 3A illustrates a full step sequence called "one phase on" or more commonly "wave drive." In this mode, a single stator coil is energized and the rotor is aligned with the stator poles.

Figure 3B energizes two coil phases simultaneously so that the rotor is held magnetically between the stator positions. This is also a full step sequence called "two phase on" or the "normal drive" sequence for a bipolar motor. Since both coils are energized, 40% greater torque is produced in this mode.

Figure 3C is a combination of the preceding sequences. It is easy to see that the normal drive will place the rotor halfway between the stator poles alternately, while the wave drive will line the rotor up with the stator poles. Interleaving the two sequences produces twice as many steps and is therefore referred to as "half-step." The combination also produces an irregular torque.

The advantage of the half- step is it

**Figure 4.** Typical Normal/Half-Step Torque Curve.



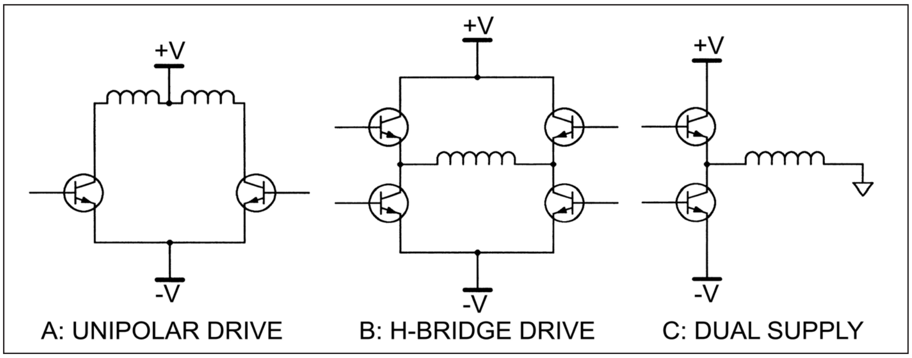



**Figure 5.** Typical Drive Circuits.

doubles the number of steps per revolution. But the essential advantage is that the motor operation is much smoother. Figure 4 illustrates a typical torque curve of a motor operating in full step and half step mode. The torque is severely reduced in the full step mode at certain frequencies. This is due to the phenomena of resonance. The rotor and the holding magnetic field form a spring-mass system that vibrates. Resonance can become so severe that the motor will completely stall and miss pulses so that all position information is lost. A load (frictional damping) generally helps to reduce this effect but it may not fully eliminate it. It is highly instructive to actually operate a stepping motor with a variable frequency pulse generator to observe this effect. Once these resonate frequencies are identified, the motor speed can usually be changed to avoid these regions. If the motor must operate at these speeds, then the motor should be switched to the half step mode in these regions.

## Control Electronics

The advantage of the unipolar motor is the simple drive circuitry, an important consideration when discrete devices are used. As illustrated in Figure 5A, only two transistors are required to reverse the direction of the current and they are easily driven because they are ground referenced. The disadvantage is the bifilar coil that is required. As already mentioned, the double coils require thinner wire to achieve a specific bulk factor that results in higher resistance and reduced efficiency. In addition, the unipolar motors are more expensive to manufacture because of the increased coil complexity.

The bipolar motor offers a single winding with the advantage of heavier wire for the same specific bulk factor and, therefore, reduced resistance with improved efficiency. The disadvantage is the more complex driving requirements. This disadvantage is mitigated by the use of highly integrated driver circuits. As illustrated in Figure 5B, four transistors are required to reverse the current. The lower two can easily be driven because they are ground referenced, but the upper two are more difficult to drive because they are on the high side of the power supply. This drive scheme is referred to as an "H-bridge" because it resembles the letter H.

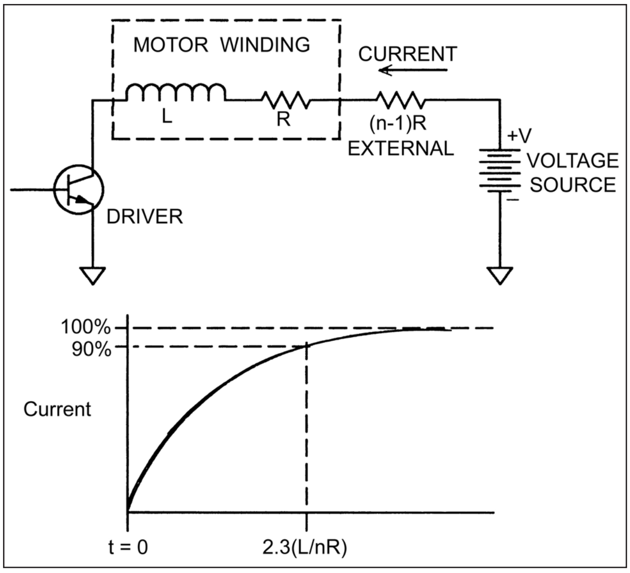



**Figure 6.** Current Rise Time for "nR-Drive.

While we are on the subject of driving the coils, we should mention that the bipolar motor can also be driven with only two transistors if a bipolar power supply is available, as illustrated in Figure 5C. There are several disadvantages to this scheme. Both transistors are difficult to drive from a ground referenced logic signal and they must be rated at twice the voltage applied across the motor winding. However, the major problem is that it requires the use of two power supplies. In addition, the two power supplies are not efficiently utilized, since only one sources current at a time.

There is an inherent difference between stepping motors and common AC/DC motors. Conventional AC/DC motors are energized from a constant voltage source. When a conventional motor starts, the load inertia prevents the rotor from instantaneously accelerating and the motor draws an initially high current to start. On the other hand, a stepping motor moves a load through a discrete step per pulse and then stops until the next sequence of pulses. During the stopped time, a stepping motor operates like a conventional motor with a locked rotor to maintain position and the current must be controlled to prevent disaster.

The simplest method to achieve this is to apply an appropriate voltage such that the coil resistance will limit the current to a safe level. A generalized drive circuit for a coil is illustrated in Figure 6. For the following discussion, assume that the external resistor (n-1)R =0 and the switching transistor is off at time zero, so the initial current will be zero. When the switching transistor is turned on, the current increases exponentially, as described below:

$$I_{(t)} = \frac{V}{R}\left(1 - e^{-\left(\frac{R}{L}\right)t}\right)$$

The time constant τ (seconds) for this circuit is simply L/R. The current will reach 90% of the steady state value in 2.3 time constants. For a typical motor with 1.8° per step (400 steps/rev), L=16 mH and R=15 ohms, the motor would reach 90% of its final torque in approximately 2.5 mS. This response time limits the maximum rotational speed to less then 60 RPM. This is only a rough calculation, since the time for the current to reverse and decay is not included. Also, the coil inductance is not constant and changes with rotor position, which affects the current rise time. In addition, the average current will be about half this value because it has to ramp up towards a steady state value, so that less torque is produced. If we try to pulse the motor faster, the current will never reach a steady state value and the motor produces little torque and may even fail to move. This process is also illustrated in Figure 4 and is responsible for the drop in torque at higher speeds.

Notice, however, that the time constant can be reduced for larger values of R. If we add an additional external resistance (n-1)R, it will improve the response. However, as the external resistance increases, we must also increase V to maintain the same current for a given motor torque. This drive scheme is often referred to as "nR-drive" since the reciprocal of n is also the improvement in the time constant. The price you pay for increased speed by this scheme is increased power dissipation in the dropping resistor during steady state conditions. Also, since additional power must be supplied, the cost of the power supply increases.

## Power Management

nR-drive suggests that stepping motors should be driven from a constant current supply instead of a constant voltage supply. Indeed, this is true and constant current supplies will work. However, they present the same problems as nR-drives. In steady state condition, the current source must dissipate the additional power, as well as the power supply. Because the current in a stepping motor must be controlled, the voltage applied is often greater than the voltage listed on the specifications. It is not unusual to see a 5V stepping motor operating on a 24V supply, whereas you would never want to operate a conventional AC/DC motor on a supply greater than specified on the nameplate.

Alternately, the time constant can be improved by reducing the coil inductance L. A typical unipolar motor will have six leads, while a typical bipolar motor will have four leads. Some motors have multiple windings with eight leads that allow their use in either unipolar or bipolar applications. Other motors are designed with multiple windings that can be placed in either series or parallel connections. The total inductance of two identical coils in series combines like resistors in series and is just twice the value of an individual coil. Well not quite, you remember the caveat? This simple formula only applies to two identical coils that are far removed from one another or shielded so they don't interact magnetically.

The general equation for the inductance of two tightly coupled coils in series is given as:

$$L_t = L_1 + L_2 \pm 2M$$

where M is the mutual inductance due to magnetic coupling. In the case of the stepping motor, the coupling is tight with $L_1 = L_2 = M$ and the mutual inductance is additive. Thus, the total inductance of two identical coils tightly coupled in series is 4L.

The general equation for two coils in parallel with magnetic coupling is given as:

$$\frac{1}{L_t} = \frac{1}{L_1 \pm M} + \frac{1}{L_2 \pm M}$$

In the case of the stepping motor, again $L_1 = L_2 = M$ and the mutual inductance is additive. Thus, the total inductance of identical coils tightly coupled in parallel is just L.

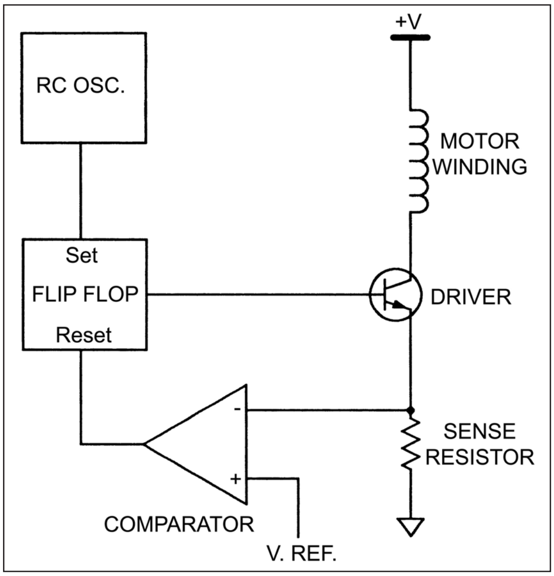



**Figure 7.** Chopper Constant Current Motor Drive.

All of the methods described have been employed to increase the drive speed of stepping motors but at the expense of additional power dissipation. To avoid this power penalty, the chopper current control has been borrowed from the switching power supply design. This approach is illustrated in Figure 7. The current through the motor winding is sensed across a low value resistor and compared to a reference value. The comparator pulse width modulates (PWM) the pass transistor in order to maintain an average motor current. The result is improved current rise time and improved motor torque at higher speeds with high efficiency. The effect can be as high as an n-drive where n is about five (the power supply is also five times higher) without the power dissipation lost in an external resistor or the power supply. The separate oscillator makes the chopper frequency independent of external components, especially the motor inductance.

## IC Selection

While the stepping motor pulse sequence can be generated by discrete logic or a microprocessor, there remains the current drive and chopping arrangement to consider. Most often, the best solution to all these problems can be obtained from a dedicated controller IC and driver IC. All of these elements discussed and more are designed into a

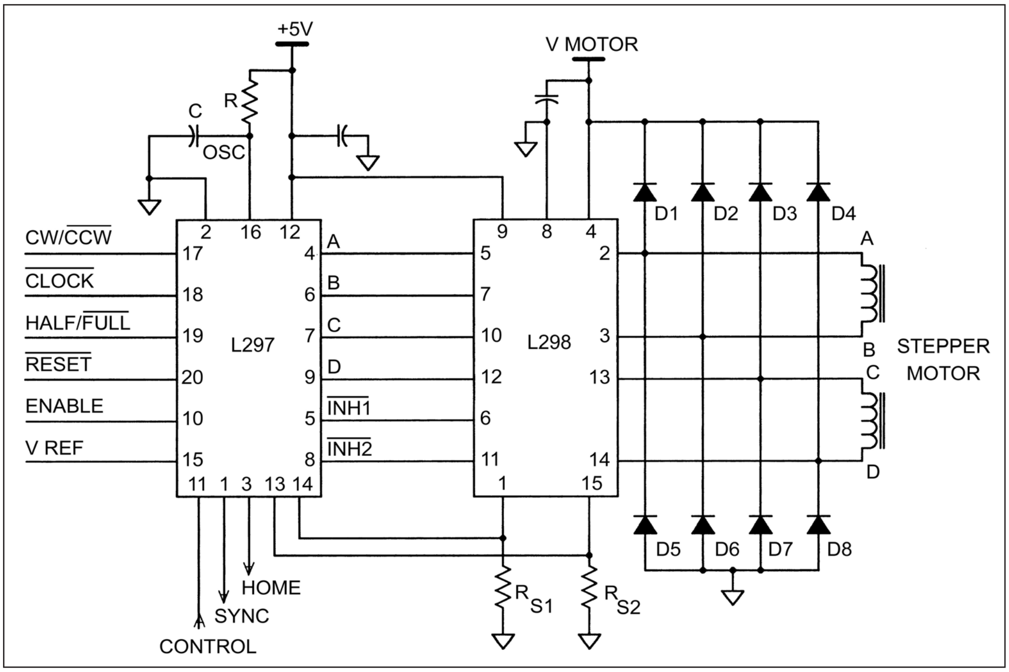



**Figure 8.** Typical Configuration for the L297/L298 Motor Drive.

values as low as 0.1Ω can be used. The lower value resistors will dissipate less power and remain stable. Higher values will provide a larger signal if excessive noise is a problem.

The eight diodes must be designated for high speed switching (less than 200 nS) in order to clamp voltage transients from the inductive motor winding and protect the driver transistors. Suitable 3A diodes are FR302, MR851, and 1N5822 (Schottky rated to 40V). The 1N4935 can be used with motors up to 1A. Never use a conventional rectifier like the 1N4001 as a clamp diode. The diode chosen should have a switching speed the same or better than the transistor and a reverse breakdown voltage greater than the motor supply voltage. In addition, it should be able to conduct a forward current as large as the maximum motor current.

number of integrated circuits from various manufacturers. Data sheets can be obtained directly from the manufacturers or through their websites. Some popular stepping motor ICs are the L297, L298, L6202, L6203, L6207, L6209, L6210, L6217A, L6506, MC3479C, and the PBL3717A. (Don't confuse the L297 with the L297A — the latter incorporates a special pulse doubling circuit intended to drive floppy-disc head positioning motors.) Typical among these devices is the venerable L297 controller and the L298 driver manufactured by ST Microelectronics (**www.st.com**). These two chips can provide all three basic pulse sequences with current chopping for a bipolar motor as illustrated in Figure 8. A printed circuit board and kit of all the parts is available from Kevin Ross at **http://kevinro.com** as K-CHP-KIT for $39.95 plus shipping.

The L298N is supplied in a 15-pin dual row package that mounts vertically to a heatsink. As seen in Figure 8, the L298 driver IC simply interfaces between the motor controller and the motor. All voltage level shifting is handled internally. The H bridge switching transistors and logic have been designed to prevent simultaneous conduction between devices located on the same leg to prevent rail-to-rail shorts. Also a small dead time is allowed during which all transistors are off. A thermal protection circuit is included that will disable the transistors if their junction temperature becomes excessive. The motor voltage, pin 4, can be as high as 46V or as low as 5V. Voltages below 5V will not harm the driver, but the drivers will not operate properly. When properly heatsinked, the driver can run 2A per motor winding. Higher current output can be obtained by paralleling the drivers of one device together. This allows operation of 3.5A for one motor winding and an additional device is required for the other motor winding. To insure proper current distribution in a device, driver A (1) is paralleled with driver D (4), and driver B (2) is paralled with driver C (3).

The maximum comparator reference voltage is 3V at the maximum driver current of 2A which would require a sensing resistor of 1.5Ω at 6W. However, the driver is designed so that the motor current only flows through the sensing resistor while the drive transistors are on. When the drive transistor switches off due to the chopper action, the current flowing in the winding is shunted by the switching diodes. If the chopper maintains an approximate 50% duty cycle, then the actual sense resistor could be half this size. Typically, a resistor of 0.5Ω at 1W is used with currents of about 2A, but

## Utilizing the L297

The L297 is packaged as a 20 pin DIP. The various control lines of the L297 control IC will be discussed next. A total of 10 pins are used to control or program the chip. All inputs and outputs are fully CMOS/TTL logic compatible. The L297 can be used with any form of power stage and will provide 20 mA drive current for this purpose. It is good practice to place a small ceramic capacitor in parallel with a small tantalum or electrolytic capacitor near the chip to filter the power supply voltage.

The RESET line — pin 20 — is normally pulled high (+5V). A low on this line will reset the internal translator logic to the home position with A, B, C, and D = 0101.

FULL STEP mode (this includes the normal drive and the wave drive) or the HALF STEP mode is selected on pin 19. If pin 19 is tied low and pin 20 is tied high, upon power-up, the controller will be reset with the translator home (A, B, C, and D = 0101) and ready to generate the normal drive mode. Similarly, if both pins 19 and 20 are tied high, upon power-up, the controller will generate the half step mode. To enter the wave mode is more difficult because it is seldom used. In order to enter the wave mode, set pin 19 high with ENABLE, pin 10, set low in order to disable the output stage and prevent motor movement. RESET, pin 20, is brought low momentarily (or on power-up) to enter the home (0101) position of the translator. Next, one clock pulse must be executed to position 0001 (or 0100 in the opposite direction), and then bring pin 19 low and return ENABLE, pin 10, high to allow the wave mode sequence to be executed. If you wish to switch between normal drive and half step mode while the chip remains powered up, it is advisable to set ENABLE, pin 19, low to first inhibit motor movement. Then momentarily bring RESET, pin 20, low to reset the translator to home position and then set pin 19 to the desired state. Finally, return Enable, pin 10, high to allow motor operation.

CW/CCW motor direction is controlled by pin 17 and is relative to the phasing of the motor windings.

CLOCK pin 18 will accept a standard CMOS/TTL pulse to step the motor. It will advance the pulse sequence on each rising edge. For example, the rising edge will advance a 1.8° stepping motor one step in the normal drive mode and 0.9° in the half step mode. The clock frequency (f in Hz) is related to the motor RPM as:

$$RPM = 0.166 \times \frac{angle}{step} \times f$$

ENABLE, pin 10, when low prevents motor operation by setting A, B, C, D, INH1, and INH2 all low.

$V_{ref}$ established on pin 15 sets the dual voltage comparators for the maximum voltage across the two current sensing resistors and establishes the maximum motor winding current. The maximum voltage is specified at 3V. The comparator has a maximum offset voltage of 5 mV. The comparator bias current is rather large but of little concern for low impedances as is the general case.

HOME, pin 3, is an open collector output pin that sinks current when the translator is out of home position (A, B, C, and D not equal to 0101). A pullup resistor can be connected to this pin and the +5V supply to furnish a logic level signal. HOME position is then signified by a high level signal (A, B, C, and D = 0101).

SYNC, pin 1, can be used to synchronize multiple controller chips

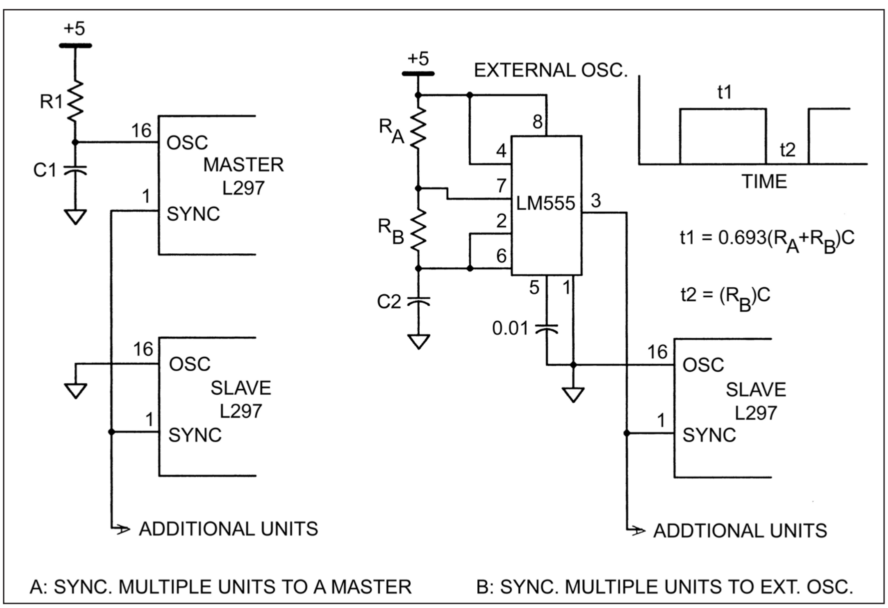



**Figure 9.** Sync. Multiple Units.

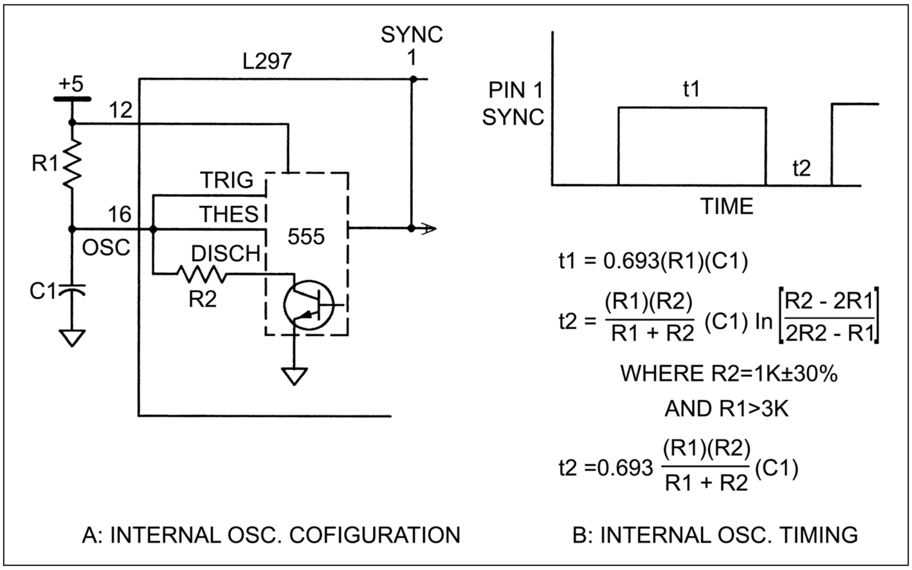


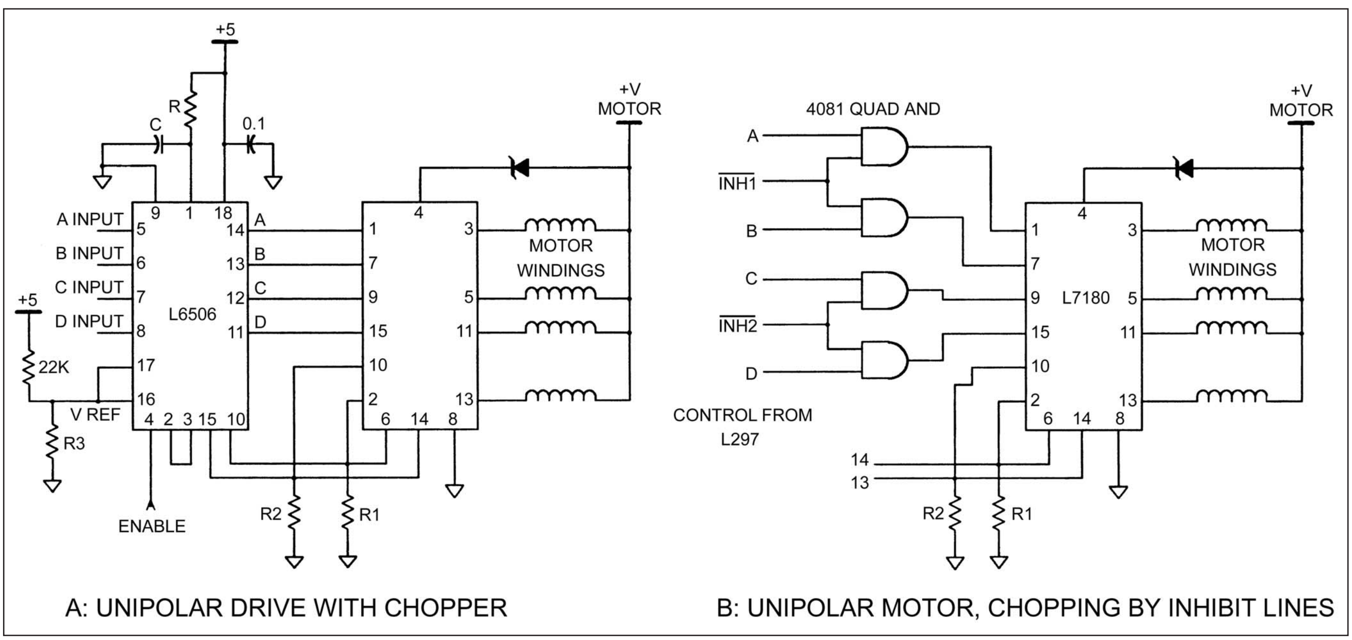

**Figure 10.** L297 Internal Oscillator.

to the same frequency to avoid excessive noise spikes. This can be achieved by connecting all the controllers together as illustrated in Figure 9A. Alternately, if the chopping frequency is furnished from an external source, it can be connected as shown in Figure 9B.

OSC, pin 16, sets the chopper oscillator frequency by means of an R/C network. The internal oscillator is similar to a typical 555 timer with a fixed internal 1K discharge resistor as illustrated in Figure 10A. While the RC product determines the oscillator frequency, it is important to consider the impedance of these components. As can be seen in Figure 10B, the ratio of R to C will affect the internal oscillator duty cycle. While the oscillators illustrated in Figure 9B and Figure 10A use the same basic timer, the value of the discharge resistor is fixed in the latter. This affects the range of timing of t2. While t2 is held low, the chopper latch operation is delayed to avoid parasitic current spikes from affecting the true current sensing. The L298 has a propagation delay of about 3 µS plus the switching time of the clamp diodes of an additional 200 nS so that the minimum value of C is 0.005 µF. For a chopper frequency of about 25 kHz, R = 10K. The manufacturer recommends the minimum value of 10K for R. Generally, the chopper frequency is chosen above the normal audio range. Longer time for t2 is available from the circuit of Figure 9B or alternately two low pass filters (four components) can be installed between the sensing resistors and the L297 sensing inputs to reduce the effects of parasitic current spikes. If the chopper function is not employed, the SENSE input, pins 13 and 14, must be grounded and $V_{ref}$, pin 15, should be tied to ground or the +5V to prevent noise pickup.

CONTROL, pin 11, defines the operation of the chopper. When set low, the chopper will act through the two INH lines. When set high, the chopper operation will inhibit the phase lines A, B, C, and D. Generally, the chopper is allowed to operate through the INH lines. The INH lines switch the respective H set of driver transistors all off or to a high impedance state. In this mode, the motor current is allowed to flow through a lower clamp diode and an upper clamp diode to the power supply. This is a very fast decay route for the motor current. The decaying current is also shunted away from the current sensing resistor, thereby reducing power dissipation

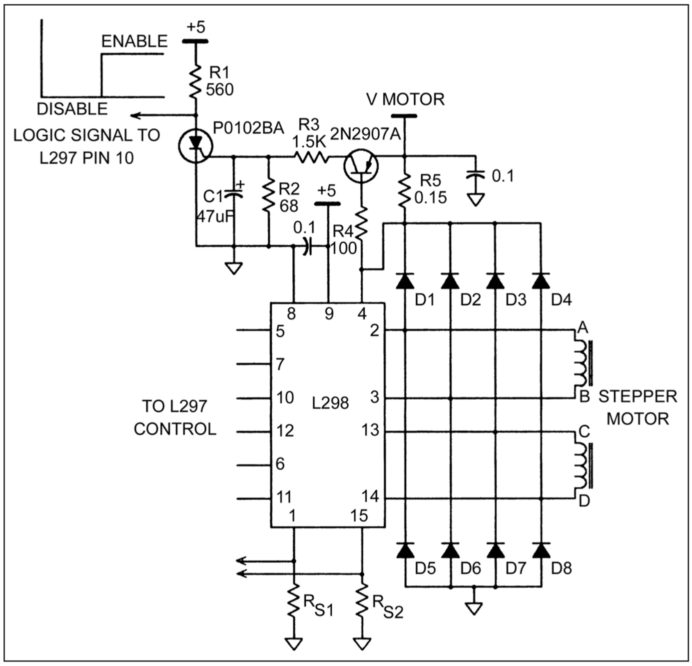
**Figure 11.** Unipolar Motor Drive.



in it. It should be mentioned that the INH lines are always active during the half step and wave drive mode, regardless of the CONTROL state. The reason for this is that in both of these modes, there is a time when the motor current is shut off and the current must decay quickly. Chopper action by setting CONTROL, pin 11, low is the only choice for unipolar motors, since one winding is always inactive. The INH signals are not generated during the normal mode sequence since both coils are always energized, unless the chopper option is selected through these lines.

If the CONTROL, pin 11, is selected high, chopper action is applied through the phase lines A, B, C, and D. If, for example, at the step when A (C) is driven high with B (D) driven low and the maximum current limit is reached, the chopper logic will pull B (D) high to cut the current off (both sides of the coil are high). The current decay circulates through the upper transistor A (C) and the upper clamp diode on B (D). This path is rather slow but again the motor current is shunted away from the current sensing resistor to reduce power dissipation. This option is useful with very low inductance motors that don't store much energy. If a rapid current decay rate was selected with such a motor, the average current may become too low for the motor to provide useful torque.

**Figure 12.** Drive Shaft Circuit Protection.




## Unipolar Drive

Figure 11 illustrates some drive concepts for unipolar motors, which can also be driven in the same fashion as a bipolar motor. Half the motor winding can be excited with an H-bridge drive at the normal unipolar current rating or the whole winding can be excited with an H-bridge drive at half the unipolar current rating. If the latter method is employed, remember the coil inductance is now four times as large as a single coil and for this application, never connect the center tap to ground or a voltage source.

The H-drive requires the coil to be floating for proper operation. If the motor winding becomes shorted, the current sensing resistor will protect the driver by means of the current limit. Similarly, if the output lead becomes shorted to the power supply, the driver will be protected. The worst case is a motor winding to ground. This will cause excess current and will undoubtedly destroy the driver before the internal thermal protection can protect the device. If your application may have a coil short to ground, at least provide a fast-acting fuse in the power supply lead to the driver chip. Faster acting electronic short circuit protection can be added, as illustrated in Figure 12. Current sensing resistor R5 in the driver power supply lead is used to sense a fault. When approximately 0.65V is generated across this resistor, the PNP transistor is forward biased and switches on. This fault detector triggers a sensitive gate SCR to disable the driver stage and it remains tripped until the power is cycled. The exact values for the current sensing resistor and attenuator (R3, R2, and C1) may be adjusted for a specific application. In place of the SCR, another transistor could also be used to control the L297 ENABLE, pin 10.

The high frequency and relatively high voltage pulses that drive the stepping motor windings often lead to radio frequency interference (RFI) that can be radiated by long motor leads to affect other more sensitive circuitry. Leads should be twisted and made as short as possible to reduce this problem. Sometimes it is helpful to wrap several turns of the lead wires around a ferrite (a high frequency magnetic material) toroid. The toroid should be located as close as possible to the driver chip. The effect of the toroid is to increase the inductance of the leads and reduce high frequency radiation. Good construction practices will also help, especially a clean layout with a good ground plane and power supply bypass capacitors near all the integrated circuits.

Now that the basics have been covered, we can deal with some circuit variations and improvements. During a positioning operation, more energy is required due to the acceleration and deceleration. While the motor is at rest, only a small holding torque is required. If the motor current is large for the positioning operation, power is wasted during the holding time. For such a situation, it is relatively easy to modulate the Vref to satisfy both conditions since the peak motor current is equal to $V_{ref}/R_{sense}$.

Modulation of the $V_{ref}$ can also be utilized to obtain micro stepping capability. We have already seen how half-stepping occurs. The half-step sequence includes the normal drive sequence in which both coils are energized with the rotor positioned halfway between the stator poles. This position is generated by equally energizing the two windings. If one coil were fully energized while the other was only half the strength, a new position of the rotor would be established — approximately a quarter of a step. Quarter-stepping yields 800 positions per revolution for a 1.8° motor. This is the concept of micro stepping. As you can imagine, this can become very complicated and expensive. Large scale integrated (LSI) circuits and microprocessors make this possible. A microprocessor with two internal eight bit A/D converts can be used to establish two independent $V_{ref}$s for each winding. The motor currents approximate a sine wave and the motion of the motor becomes very smooth and gentle. This requires a control chip with two comparators, like the L6506 or complete micro stepping controllers, such as the L6217A or PBL3717A.

## Conclusion

As discussed earlier, the half-step provides greater smoothness and resolution, but it provides 40% less torque than the normal drive mode. It is possible to compensate for this torque loss in the half-step mode. Since the torque is proportional to the motor current, a 40% increase in the motor current of the active winding can compensate for this effect. If you look at the half-step mode, you will realize that the torque loss is due to the wave mode in which only one winding is energized. The motor current rating for a bipolar motor, however, always refers to operation with current in both coils (normal drive mode). A 40% increase in a single winding while the other winding is inactive will just double the power to the normal drive mode level. It is, however, inadvisable to stop the motor in a position where a single coil is receiving 40% more power. This situation differs from running the motor since the single coil must sustain the self-heat, unlike two coils in the normal drive mode. As illustrated in Figure 13, the phase line pairs are decoded to detect the time when A, B = 0 or C, D = 0 and the reference voltage is increased by 40%. The values illustrated produce a normal reference value of 1V and a boosted value of 1.4V. Motor current can be adjusted by selecting suitable current sensing resistors or proper redesign. In addition, logic control of the Torque Boost is available to reduce the motor current when the motor stops. If the motor is operated at a fixed pulse rate, a missing pulse detector can be added to provide this logic function automatically. **SV**

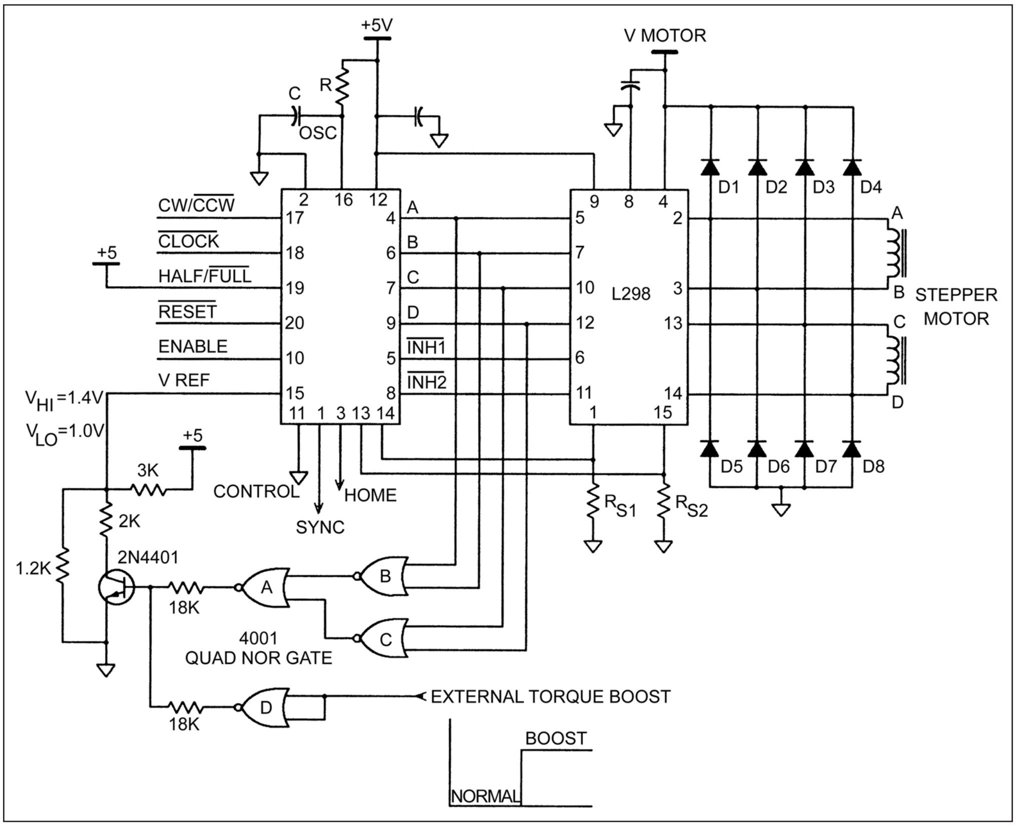



**Figure 13.** Improved Torque in the Half-Step Mode.

[Part 1]

# INTERMEDIATE ROBOTS

## Building a Laptop- or PDA-Based Robot

**MEET THE 'BOTS**

- HelmBot — iPaq PDA Robot (left)
- Seeker — Laptop-based Robo-Magellan Robot (below)

If you reach the point where an embedded processor just won't provide the processing power and features you want for your next robot, the addition of a laptop or PDA into your design may provide just the boost you need. This article is the first in a mini-series describing how to build a high-performance laptop- or PDA-based robot for a surprisingly reasonable price.

For this first article, I will focus on the electronic hardware design of Seeker — the winner of the 2005 SRS Robo-Magellan competition. Seeker is a laptop-based robot built onto an RC truck chassis. In a future article, I will describe the differences between this and HelmetBot — an iPaq-based Internet-controllable remote-presence robot that is built into a bicycle helmet. Both robots use similar hardware and software, with changes as needed to fit their design goals.

B Y  D A V E  S H I N S E L

## Why a Laptop?

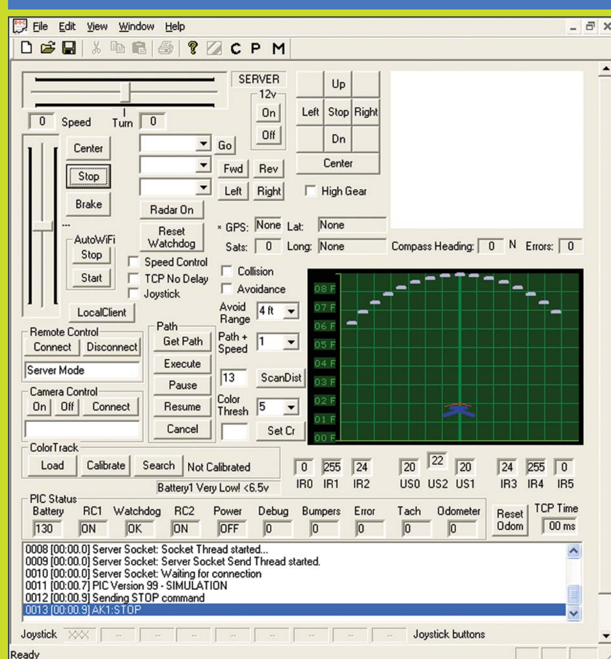
Seeker was designed to be able to navigate outdoors, following complex paths while avoiding obstacles and spotting landmarks. (Visit the Robo-Magellan webpage at **www.robothon.org** for more information.) In addition, it was highly desirable to be able to monitor, control, and debug the robot from a remote computer. Further, I wanted to do vision processing, which meant that I needed some reasonable computing power. I had been looking into several single-board computers for the job, when I discovered that you can buy a 300 MHz laptop on eBay for around $100! You don't realize how many great features are packed into a laptop until you try building a robot with a single board computer! In one fell swoop, I got plenty of processing power, memory, hard drive storage, power supply and battery, keyboard, display, and USB. For another $15, I added an 802.11 PCMCIA card, and had instant wireless.

## Capabilities

Seeker provides a lot of intermediate to advanced capabilities, but is easy enough to be built by anyone with basic knowledge of C++. Even if you don't have experience in C++, this could be a great opportunity to learn the basics of this flexible language. In this series of articles, I hope to provide you with enough information so you can build a robot with:

- An interface between a PC/Laptop/PDA and embedded microcontroller
- Sensor fusion of IR sensors, ultrasonic rangers, electronic compass, and GPS
- Object avoidance and behavior programming
- Control and sensor display Graphical User Interface (GUI) (Figure 1)
- Path entry for following a complex course (Figure 2)

**FIGURE 1. Seeker Command GUI**



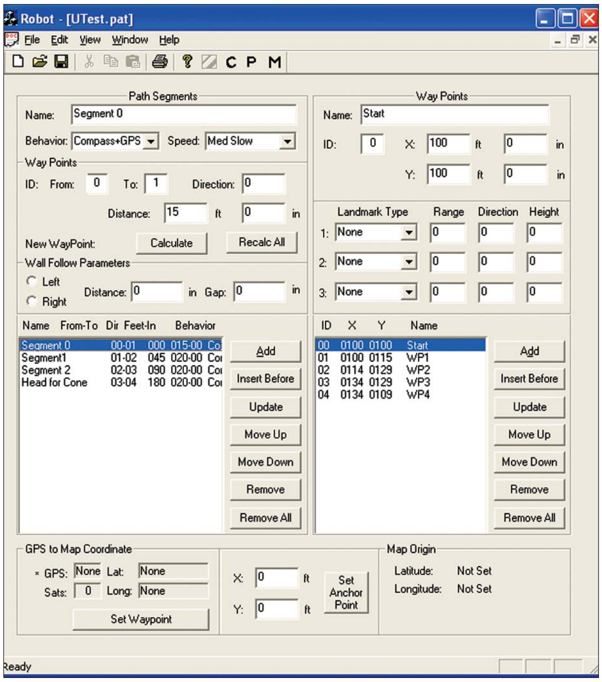



FIGURE 2. Seeker Path Entry

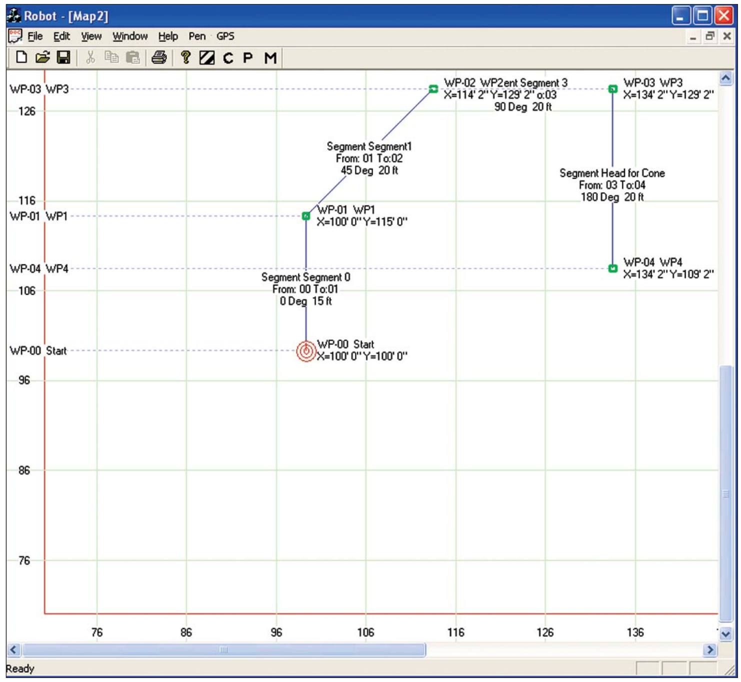



FIGURE 3. Seeker Map

■ Hierarchical behavior processing
■ Graphical mapping for navigation debugging (Figure 3)
■ Wireless remote operation and debug from another computer

## Hardware Design Overview

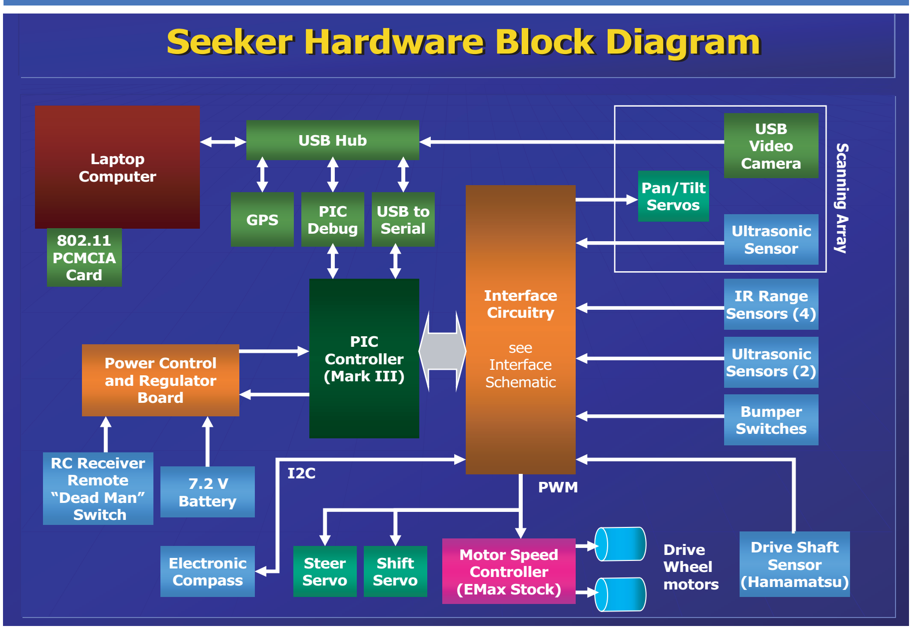
Look at Figure 4 — the laptop handles the higher level functions of the robot, and communicates through a USB hub to a:

■ USB video camera
■ USB PIC debugger (optional)
■ PIC microcontroller (via USB to RS-232)
■ GPS (optional)
■ 802.11 wireless (USB or PCMCIA card)

The low level, real-time tasks and hardware interfacing is handled by the PIC microcontroller. The laptop communicates with the PIC via RS-232. I choose to use a USB-to-serial converter to minimize the connections to the laptop, but a direct serial cable from the laptop works fine, as well.

The PIC handles these low-level tasks:

FIGURE 4. Seeker Hardware Block Diagram




■ Ultrasonic echo timing and calculation
■ IR sensor reading and scaling
■ Bumper switch reading
■ I2C compass reading
■ Battery monitoring
■ Drive shaft odometer and motor speed feedback (closed loop)
■ Pulse Width Modulation (PWM) for servo control (steering, speed, gear shift, camera pan/tilt).

## Power Supply

Power for all electronics is provided by a standard seven-cell NiMH RC car power pack, which delivers up to 8V. I started with a six-cell pack, and found that the electronics became erratic as the battery discharged. With seven cells, the run time was greatly improved because the electronics run right up to the point where the battery was completely drained. The 8V is fed to the Mark III controller board, which has a small on-board voltage regulator sufficient for running most of the on-board electronics. The 8V also goes to a standard 7805 voltage regulator. The 5V output of this regulator provides plenty of current to drive all the sensors and the USB hub.

For Seeker, I used a separate battery from the battery packs used to drive the motors. The motors on Seeker draw a lot of current, so the noise spikes on the battery line were unacceptable. For most robots, I usually try to use the same battery for both.

## Watchdog and Dead-man Switch

The SRS Robo-Magellan rules require that a remote device must be used to enable power to the wheels. In addition to the remote cut-off, I wanted the PIC to have a watchdog timer, such that if the laptop ever stopped communicating, the PIC would cut the power to the motors automatically.

The quickest and cheapest way to implement this was a "poor man's" AND gate, using two readily available 2N2222A switching transistors to enable a small reed relay. As you can see from the design, both transistors must be turned on for the relay to engage (Figure 5). I placed an LED in parallel with the relay coil to indicate when the motor power was enabled. I installed override jumpers to enable the transistors directly during testing (I got tired of holding down the dead-man switch on the remote all the time).

I decided to build the motor enable circuit onto the same perf-board as the 5V regulator. For the remote control, I hacked a cheap toy airplane controller, which had two buttons on the transmitter, and a small circuit for the receiver. Using an oscilloscope, I probed the main IC on the receiver until I found two pins that toggled state along with the button presses. I wired these to the connector which plugs into the Motor Enable circuit.

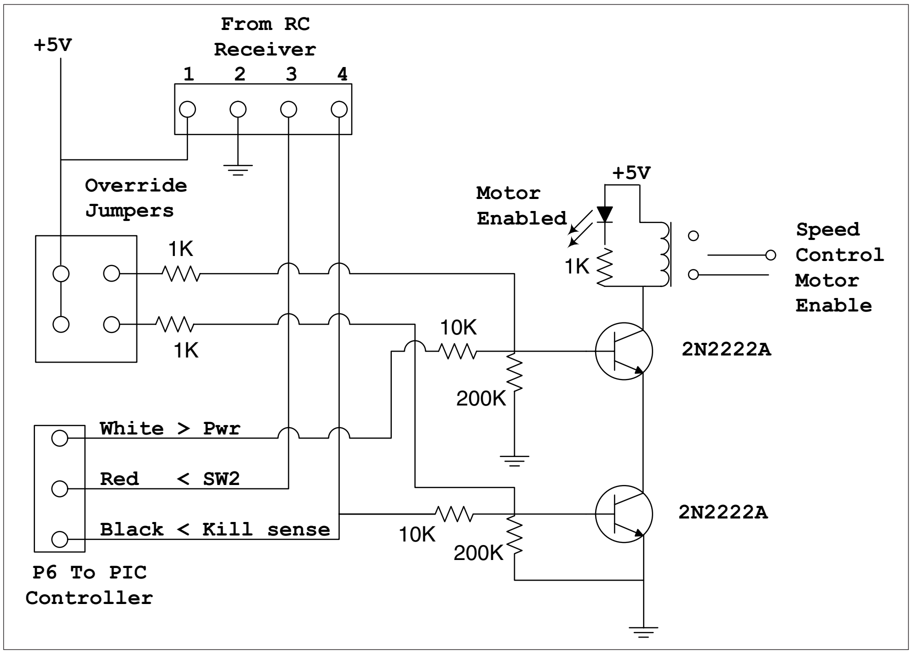



FIGURE 5. Motor Enable Circuit

## PIC Microcontroller Hardware Interface

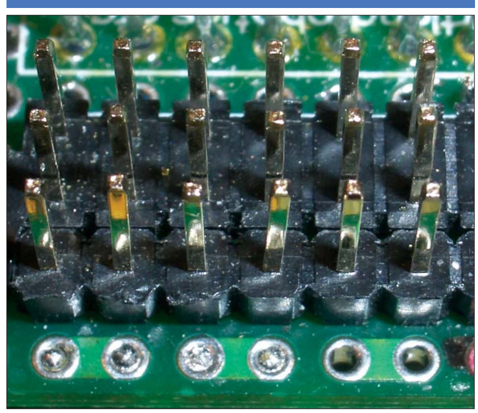
The PIC 16F877 is a common, inexpensive 40-pin controller that you can pick up for less then $8. Once loaded with a free boot loader, the PIC can be programmed over standard RS-232. This PIC has 8K of Flash memory for the program (plenty for most applications), 368 bytes of data memory, several interrupts and timers, eight A/D converters, hardware serial interface, and 33 I/O lines. There are many newer (and more powerful) controllers on the market, any of which would probably be great for this application, but I chose the PIC because of its low price and the huge support base. There are tons of examples, and a number of great PIC books available.

I used the PDX Robotics Mark III robot controller board to simplify hardware design. This board sells for about $30 (see Parts List for source), and includes the PIC pre-loaded with a boot loader. This is a nice, complete kit that has all the parts and takes less then an hour to assemble, but you should have some experience using a soldering iron.

The Mark III board has a 40 pin header for expansion. For some of my robot projects, I have used a 40 pin ribbon cable to connect the Mark III to my interface circuitry. For Seeker, I was tight on space so I ordered the Mark III expansion board, which mounts directly on top of the Mark III. This turned out to work very well; I highly recommend using this board.

The interface board is used to provide connectors to the various sensors and servos. I discovered (the hard way) that you really want to make sure all your connectors are sturdy and will not come out while your robot is bouncing around over obstacles. I think most experienced robot builders will tell you that this is a frequent cause of frustration. Whenever possible, I try to use

FIGURE 6. Servo Connectors



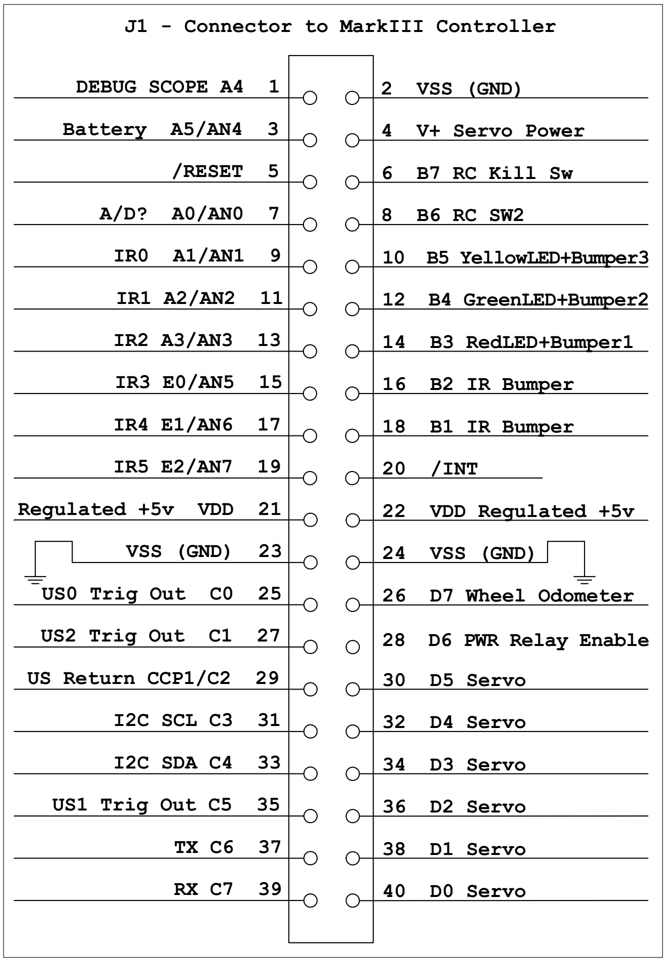



FIGURE 7. Seeker 40-pin Expansion Interface

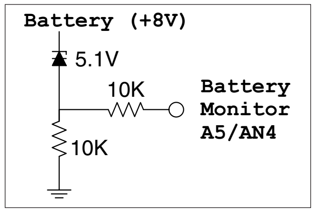



FIGURE 8. Battery Monitor

Ethernet (RJ-45) jacks for my connections. I picked up a bag of Ethernet sockets at a local surplus store for a couple of bucks. RadioShack sells the connectors and crimper tool, or you can hack up an old Ethernet cable. It's worth the investment, and the jacks plug into the sockets with a satisfying "click."

For servo and other three-pin connectors, I usually set up a row of connections so they all fit snuggly together. To do this, you need to pick up a "dual in-line header strip," plus a "single in-line header strip." Cut the header strips for the number of servos you want to connect, and then solder down the single strip next to the double strip. This gives you a row of connectors, each three pins wide (Figure 6).

## Hardware Interface Schematic

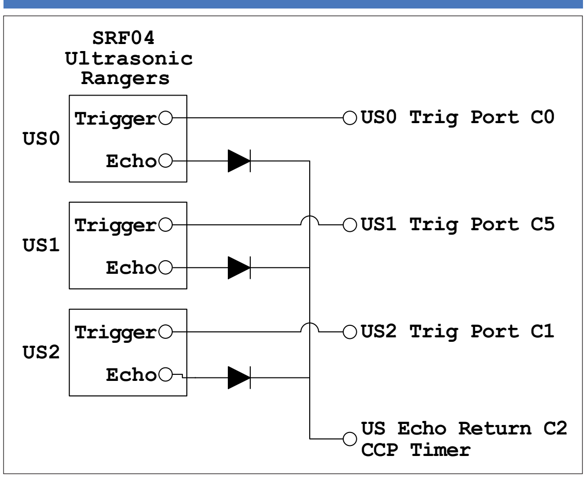



FIGURE 9. Ultrasonic Interface

For your convenience, the full schematic is available on the *SERVO* website at **www.servomagazine.com**

Figure 7 shows the functions assigned each pin of the 40 pin Mark III expansion interface.

**Starting with the odd pins on the left side of the connector:**

■ Pin 1 (port A4) is the Scope Debug pin. This pin is used to debug fast timing errors that are difficult to debug using variables or other debug techniques. For example, by connecting an oscilloscope and setting the right values in the PIC software, this pin can show the pulse width being sent to a given servo. A scope is not required for this project, but can come in handy at times.

■ Pin 3 (port A5) is connected to the battery monitor (Figure 8). Using the regulated +5V as a reference, the PIC Analog to Digital (A/D) converters can measure voltages from 0 to 5V. Since the electronics will stop working at about 6V, and the max battery voltage peaks at less than 9V, we need to scale the voltage range. By using a 5.1V zener diode, we are able to drop the measured battery voltage range from 9V-6V down to 4V-1V, nicely within the range of the PIC A/D.

■ Pins 9-19 (A1-3, E0-2) are the infrared distance sensor's A/D inputs, and connect up to a row of three pin headers which provide +5V, ground, and the output line. The output line is an analog value from 0-5V, with the voltage indicating the distance to the nearest object. The PIC software converts the voltage value to inches.

■ Pins 25, 27, and 35 are Ultrasonic Trigger outputs. These are connected to Devantech SRF04 rangers. A 10 µS positive pulse out of these lines will cause the corresponding sensor to send out an ultrasonic pulse.

■ Pins 25, 27, and 35 (C0, C1, C5) are the trigger lines for the three Devantech SRF04 Ultrasonic rangers. To avoid interference, the Ultrasonic "chirps" are sent one at a time, in sequence. A reading starts with a 10 mS pulse on this line.

■ Pin 29 (CCP1/C2) is the Ultrasonic Sensor return line for all three sensors. The sensor's outputs are "wired-OR'd" together with the three diodes shown (Figure 9). C2 is a special port, tied to one of the two CCP (Capture/

Compare/PWM) registers. The CCPs allow precise timing of a pulse width. We use this capability to measure the round trip echo time of the ultrasonic pulse.

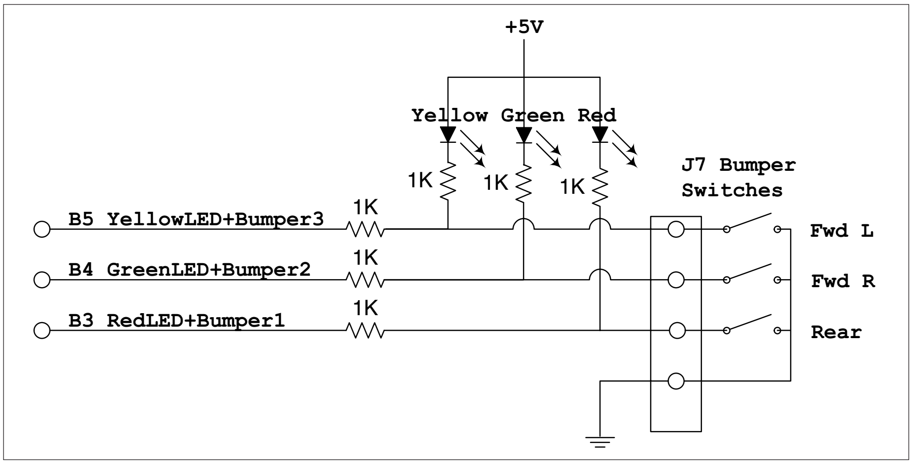


**FIGURE 10. LEDs + Bumpers**

■ Pins 31 and 33 (SCL/SDA) are the I2C Clock and Data lines. I2C is a great protocol for connecting devices together. It is currently only used for communicating with the electronic compass, but one could add more A/D lines with the MAX127 or an I2C Ultrasonic ranger such as the Devantech SRF08.

■ Pins 37 and 39 (TX/RX) are the Transmit/Receive pins for RS-232 communication with the laptop. The Mark III controller board comes with a RS-232A level shifter that converts the TTL output of the PIC to RS-232 voltage levels.

**The even pins on the right side of the connector:**

■ Pin 4 (V+) is unregulated servo power. Since servos can draw a lot of current, it is a good idea to drive them directly from the battery (not from the 5V regulator). The 8V is dropped by three two-amp diodes in series before being sent to the servos to avoid over-driving them (found that one out after frying my steering servo).

■ Pins 6 and 8 (RC Kill SW, SW2) are inputs from the RC dead-man's switch. The kill switch signal is used to monitor the state of motor power. The second switch is used as the "go" switch when starting a Robo-Magellan run.

■ Pins 10-14 (port B5-B3) do double duty, in order to conserve I/O ports (Figure 10). They serve as LED outputs for status reporting and also as inputs for reading the bumper for collisions. The software switches rapidly between output and input mode to make this work; 1K resistors protect the I/O ports and LEDs from short-circuit conditions. The green LED is a heartbeat. It blinks once each time the PIC goes through its main control loop. The yellow LED indicates serial communication. It blinks each time the PIC receives a packet of data from the laptop. The red LED indicates error conditions, such as a bad serial read.

■ Pins 16 and 18 are connected to IR "bumper switches." These handy IR sensors just switch high to low, depending if an object is within their pre-defined range. I used Sharp GP2Y0D340K sensors for this.

■ Pin 26 (D7) is the Odometer sensor input. Eight stripes of reflective tape (silver or white) are placed on the drive shaft, such that a Hamamatsu P5587 Photo-reflector sensor (Figure 11) — which is mounted next to the drive shaft — will generate 16 pulses as the shaft turns (eight high, eight low). The PIC software uses this information for two purposes: motor speed control and odometery.

■ Pin 28 (D6) is used to enable motor power. This allows the PIC to act as a watchdog, and kill power to the motors if anything goes wrong.

■ Pins 30-40 (D5-D0) are the servo outputs as indicated.

## Sensors

Sensors are increasingly more sophisticated and reliable today. Even so, each kind of sensor has its strengths and weaknesses. For example, when a robot approaches a wall from a shallow angle, an ultrasonic sensor might not "see" the wall, as the ultrasonic pulse just reflects down the length of the wall instead of back to the robot. IR sensors sometimes fail to see dark objects, such as trees, or very shiny objects, such as cones.

Here is a list of the sensors used on Seeker (Figure 12):

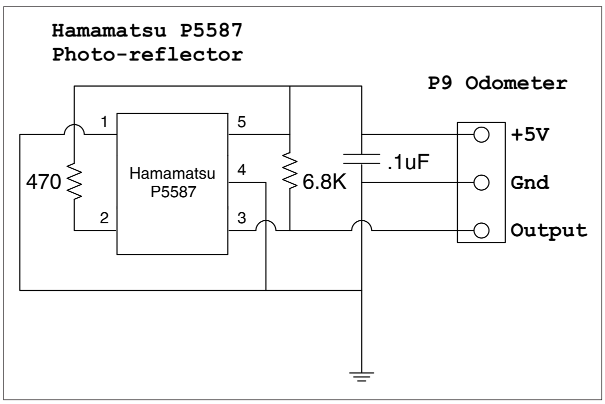
**Sharp GP2Y0A02YK Long Range IR sensor**

Two of these long-range sensors are used. They are mounted facing almost straight forward (angled in slightly) to detect objects up to four feet ahead.

**Sharp GP2Y0A21YK Wide Angle IR sensor**

Two of these shorter range, but wider angle, sensors are pointed at 45 degrees to detect objects close by on

**FIGURE 11. Seeker Odometer**



## PARTS LIST

- ☐ Traxxas E-Max or similar RC truck
- ☐ Mark III PIC Robot Controller
- ☐ Mark III Prototype Board
- ☐ Sharp GP2Y0A02YK Long Range IR Sensor
- ☐ Sharp GP2Y0A21YK Wide Angle IR Sensor
- ☐ Sharp GP2Y0D340K "IR Bumper" Distance Sensor
- ☐ Devantech SRF04 Ultrasonic Ranger
- ☐ Devantech CMPS03 Electronic Compass
- ☐ Hamamatsu P5587 IR Photoreflector
- ☐ RJ-45 Ethernet Jacks and Connectors
- ☐ USB Hub (small, 5V powered)
- ☐ 802.11 Wireless Card
- ☐ Laptop Computer
- ☐ USB Video Camera — any USB video camera should work. Look for one with a good quality sensor (some of the really small or cheap ones have "noisy" sensors).

To make three-pin servo header strips for eight servos:

- ☐ See **http://dkc3.digikey.com/PDF/T053/0144.pdf**
- ☐ (1) Dual in-line header: Digi-Key A26579-ND, AMP 87215-8
- ☐ (1) Single in-line header: Digi-Key A26555-ND, AMP 87224-8
- ☐ Or, you can get any dual and single headers, and break them to the desired length.

Mark III PIC Controller and many other parts are available at: The Mark III Robot Store **www.junun.org/MarkIII**

Most of the other parts are available from multiple sources, including:
Acroname — **www.acroname.com**
Digi-Key — **www.digikey.com**
Jameco — **www.jameco.com** or **www.robotstore.com**
Pololu — **www.pololu.com**
Devantech — **www.robot-electronics.co.uk/**

USB Hub, 802.11 wireless card, etc., are available at multiple computer stores.

Laptop computer — available on eBay
WARNING — The battery is usually the first thing to go in a laptop, and can cost close to $100. Make sure the laptop has a new battery, or budget to purchase one on the web (search for "laptop battery").

either side.

**Devantech SRF04 Ultrasonic Ranger**

Two of these are the primary sensors used by Seeker. They are mounted facing forward at about 10 degrees to either side of the center line. Since each of these sensors have a spread of 40 degrees, this provides 20 degrees of overlap where an object dead ahead will be picked up by both sensors. In addition, a third Ultrasonic sensor is mounted on the camera sensor Pan/Tilt head. This allows the robot to "look" for objects with ultrasonic pings, as well as vision. Note that other Ultrasonic sensors such as the SRF05, 08, or 10 will work as well, depending on your application.

**Devantech CMPS03 Electronic Compass**

This great little compass is used by a lot of robot builders. It is pretty

> *Shinsel built his first robot in 1980 using a 1 MHz 6502 processor and 9K of RAM.*

accurate (I see "real world" accuracy of ±1 degree), and with its I2C interface, very easy to connect and use. It is, however, very sensitive to tilt. A slope of 15 degrees can induce an error of up to 30 degrees. There are two ways to try to deal with this: a mechanical self-leveling platform, or correction through software.

**Sharp GP2Y0D340K Distance Sensor**

The output of this sensor goes low when it detects an object within about 16 inches. Since they sell for about $6, use can get several and put them wherever you need collision detection. As with any IR sensor, it can be fooled by very dark or reflective objects, so it's a good idea to have a hardware bumper, as well.

**Hamamatsu P5587 IR Photoreflector**

These handy little sensors are very tiny so they can be mounted anywhere. They provide a High/Low output corresponding to seeing black or white, and are often used with wheel encoders for measuring speed and distance.

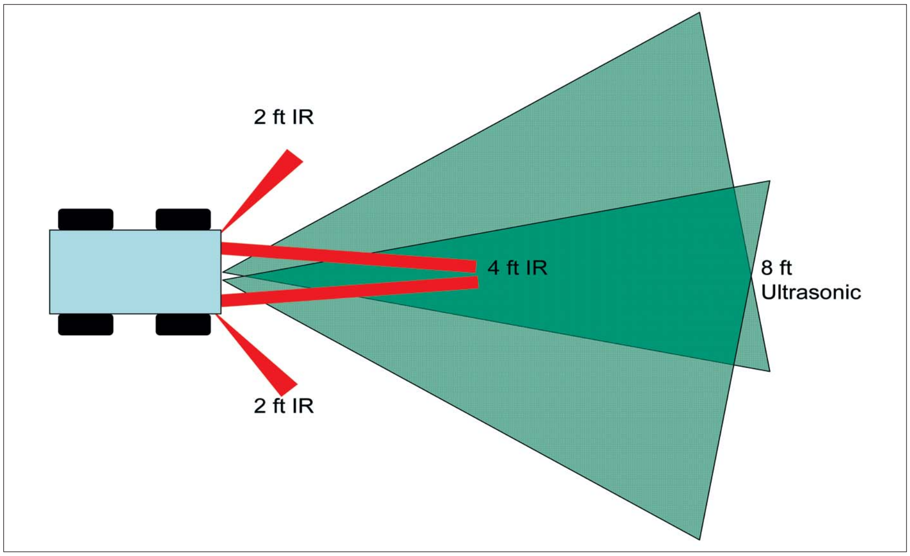



**FIGURE 12. Seeker Sensor Array**

## Next Month

In next month's article, I will go into the software design. Until then, consider this definition of hardware — "the part of a computer you can kick." SV

# Power-Assisted Cart

## Apply These Electromechanical Feedback Principles to Your Next Robotic Project

by Thomas H. Smith

This article describes a power-assisted cart built from a chassis of a four-wheel mobility scooter. The cart is shown in Figure 1 along with a "normal" mobility scooter. The cart uses a force/deflection measuring tongue to control a 24 volt DC motor to produce torque to help you move the cart. When you grab the handle of the tongue and pull, as to move the cart forward, the tongue measures the amount you pull and determines a drive signal to the motor. The motor then produces torque to help the cart move forward. The amount of torque supplied adjusts to that amount needed to maintain a constant pulling force.

If you move — as for walking forward — then the cart follows at your speed, automatically. Similar operation in reverse direction is also possible. In fact, you do not have to personally be present for the cart to operate. It can function well when used as a trailer, hitched to some "tractor" or to another such cart. The scooter in Figure 1 has a trailer hitch added at its back end so that it can pull the cart. When used as a trailer, the cart has the advantage that it does not add a significant pulling load to the tractor device: the cart pulls its own weight (and that of its load). The cart in Figure 1 has a load of two pails of sand, for a total of 110 pounds. I have used it successfully up to 250 pounds.

This cart is a power-assisted device in the same way that the power-steering on your car is power assisted: in your car a power-steering pump and hydraulic system work to apply torque to the car's steering system in order to reduce the torque that you feel in the steering wheel. In the cart, the motor works to reduce the pulling force that you exert to pull the cart. The cart operates with similar control forces (on flat ground) whether the load is light or heavy. The control circuitry of the cart uses an accelerometer to supervise the motor torque so that smooth acceleration and deceleration are obtained.

The force/deflection measuring control tongue and the accelerometer are sensors that provide feedback to the motor controller. Thus, the cart is an electromechanical feedback control system.

As a construction project, the power-assisted cart is a method of re-applying a used mobility scooter to another purpose and it is a chance to learn something about feedback control systems.

Just imagine the possible applications with your next robot.

■ FIGURE 1: Power-assisted cart and a mobility scooter.

## Operation of the Cart

The power-assisted cart is a servo system that automatically controls drive torque to adjust the speed of the cart to match whatever speed that you, the operator, are moving.

The servo block diagram of the system is shown in Figure 2. Input to the loop is change in position (motion) of the pulling agent: you (the operator), or a tractor. Output from the system is cart displacement. There are two main feedback paths shown in the figure: a position feedback for cart position relative to operator/puller position and an acceleration feedback derived from an accelerometer mounted on the chassis of the cart. Gains and frequency response shaping are present for both feedback paths; these may be seen in the schematics of Figures 7, 9, and 10.

The forward path of the block diagram of Figure 2 includes the signal processing electronics and the PWM (pulse width modulation) power amplifier. The motor, gears, and wheels form the mechanical plant portion. The wheels convert angular torque to linear forces that provide linear accelerations. The block diagram includes two natural integrators: one for acceleration to velocity and one for velocity to position.

A positive-center, spring return assembly and a slide pot form a linear position sensor mounted on the cart's tongue. The spring assembly and slide pot are connected to the pulling ring or handle by a push-rod. The spring-return system provides a resisting force that is proportional to the extension of the pushrod relative to the tongue. The spring return assembly has been built in several different configurations; these are shown in Figures 3 and 4. Displacement of the linear position sensor is determined by how much you pull or by how much you move relative to the cart. The displacement increases as you move and decreases as the cart reacts. Thus, the linear position sensor forms a position feedback sensor that measures the difference between your movement and the cart's movement. The spring system in Figure 3 provides 2.5 lbs of restoring force for a maximum extension of 1.1 inch. The extension may be either in forward or reverse direction.

An Analog Devices ADXL202 accelerometer provides a measurement of cart acceleration. The acceleration loop is added to help stabilize the system. If you operate the cart without the acceleration feedback, you will find that control settings that produce nice response for a small load mass will provide sluggish response with a heavy load. If the control system is adjusted for a responsive operation with a heavy load, then the cart tends to surge into fore/aft oscillation for a light load. Addition of the accelerometer provides a damping of the oscillations. From a practical viewpoint, the acceleration feedback signal acts to keep the speed of the cart from changing: it resists the quick changes in speed that are part of the oscillation process. Note that "acts to keep the speed from changing" sounds a lot like a description of inertia. In fact, the operation of the acceleration feedback makes it appear to the operator that the cart always has a heavy load.

When gains and frequency responses of the feedback paths and the forward path are adjusted correctly, operation of the cart with a heavy load — say 250 lbs — is very similar to operation of the cart with no load. There are no oscillations. The maximum pulling force required is the 2.5 lbs needed for full extension of the control tongue. When empty, the cart weighs around 70 lbs.

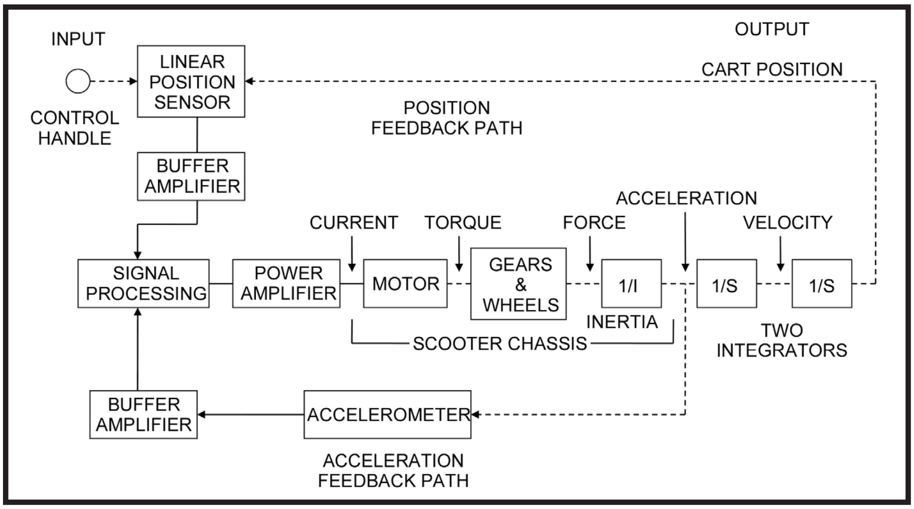



FIGURE 2: System block diagram for power assisted cart.

## The Hardware

The cart is a combination of scooter parts, home-built parts, and a commercially available PWM power amplifier. The present version of the cart has no microprocessor and no software. The cart is composed of the following major parts:

- Modified scooter chassis with 24 volt motor and brake
- Two gel cell batteries
- Load box
- Control tongue
- Accelerometer buffer amplifier
- Signal processor electronics
- Power amplifier
- Wires and cables

### Chassis

The chassis is made from a Pride

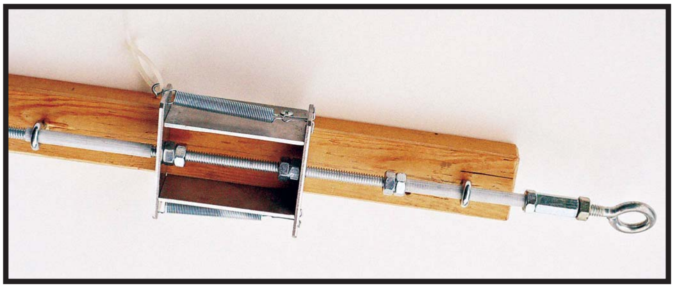


FIGURE 3: Positive center spring assembly with tension springs.

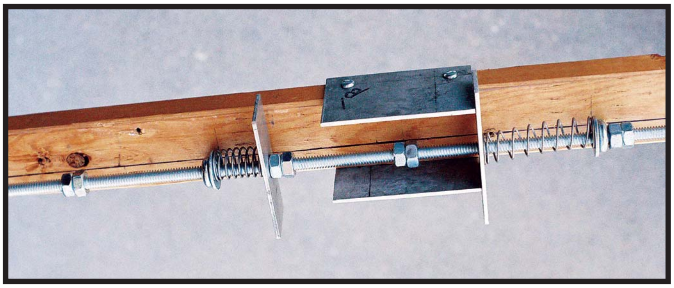
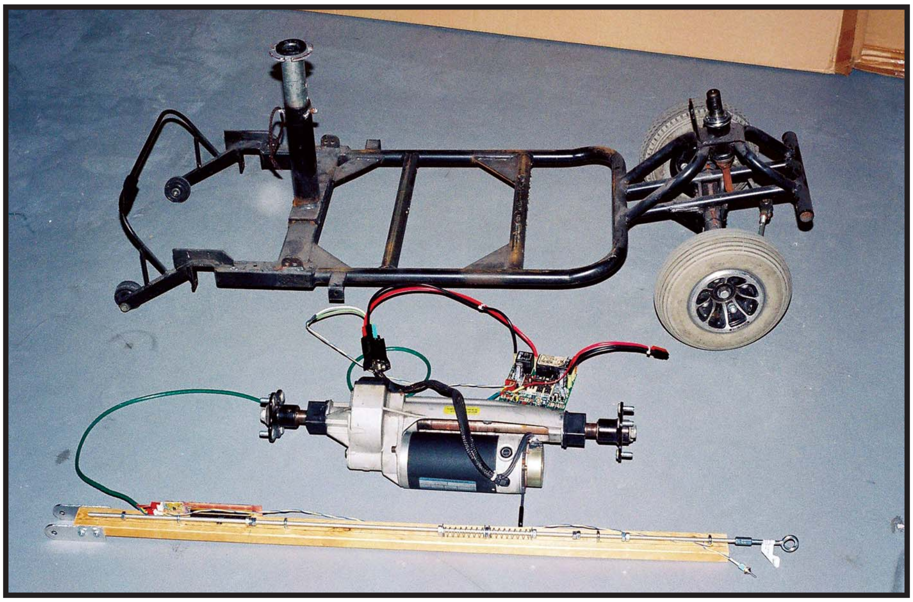


FIGURE 4: Spring assembly with compression springs, extended to one end of travel.



■ FIGURE 5: Cart frame before modifications, with motor/transaxle and control tongue.

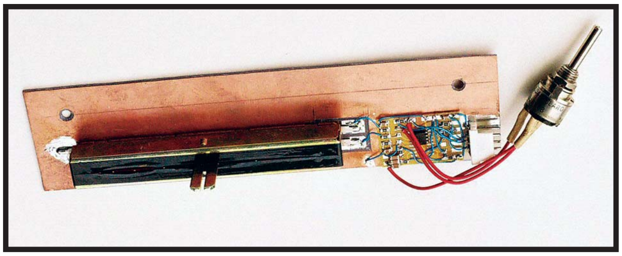
Legend scooter, made around 1990. It has a rigid frame of welded 1" black iron tubing. The chassis is a four-wheel model — but I expect that a three-wheel scooter also could be used for a load-carrying cart with a little less resistance to tipping over. Certainly, the basic size, motor power, and steering of a three-wheel model are all as suitable as those of a four-wheel chassis. The unmodified frame and the motor/transaxle assembly are shown in Figure 5.

For the cart project, all unnecessary parts of the scooter were removed: fairings, covers, steering bar, speed control pot, and so forth. The welded seat post and its flange were cut away and the site ground smooth. Later years' scooter models appear to have bolted-on seat posts; thus, the seat post will be more easily removed in those units.

I removed the steering handle for the original scooter and got down to the basic steering post that couples to the front wheels through the steering pushrods. The control tongue attaches to the steering post using a 2.5" x 5/16 bolt through a hole near the top of the post. A self-locking nut is used on the bolt to prevent loss of the bolt and consequent disaster.

## Motor

The motor on my chassis and on many other chasses is a 24 volt DC brushed motor. The motor may be rated for three amps or more of continuous current and probably 10 or more amps of surge current. The motors are usually face-mounted to a cast-aluminum gear housing. The gear housing contains a speed-reduction gear and — all-important — a differential gear assembly. The differential gear feature allows for differing drive-wheel speeds when steering through a turn, just as on an automobile. The motor and rear axle assembly are very robust. The gears are made of steel and have faces about 3/4-inch wide. You are unlikely to damage these gears.



■ FIGURE 6: Slide pot assembly with its circuit board and response-adjustment pot.

The brake on the motor is part of the safety gear for the mobility scooter — which I removed. A future version of the cart would again incorporate the brake for safety reasons

The load box is what carries the freight, or the payload. The load box for this project was made from 1" x 4" pine boards and 3/8-inch thick plywood, 2 ft square. There are 1" x 4" wood stringers attached crossways on the underside of the plywood. The floor of the box is clamped to the pair of main fore-aft tubes of the chassis. The clamps consist of four pairs of 1/4-20 bolts going through the floor and crossways stingers past the chassis tubes, then through four tie-plates on the underside of the chassis tubes.

The size, shape, and complexity of the load box is to be determined by your requirements and imagination and by the particular features of your used scooter chassis. Remember to provide for mounting and retention of batteries and electronics modules.

I use a pair of 18 Ah gel cell batteries to power the cart. These gel-cells have been adequate so far. The cells can be Power Sonic type PS 12180 NB, or similar.

## Control Tongue

The control tongue incorporates a positive-center spring return system and a linear displacement sensor, as pictured. The mechanical parts of the tongue are a 1" x 2" x 36" wood piece, a piece of 1/4-20 all-thread, and screws, nuts, and screw-eyes from the local hardware store. Several parts are made from sheet aluminum or iron straps. My local hardware store had iron straps, 1/8" x 1" x 36"; several of the parts can be made from this strap. The 1/4-20 x 2" screw-eye at the leading end of the tongue is attached to the all-thread push-rod with a 1/4-20 coupling nut — also a hardware store item. Every nut on the pushrod is held in place by thread-locking compound or by a jam-nut. You do not want to have things work loose. You may want to place part of a plastic soda straw over the push-rod threads where the pushrod runs through the screw eye guide rings; this will reduce the noise made by the threads running back and forth through the guide rings.

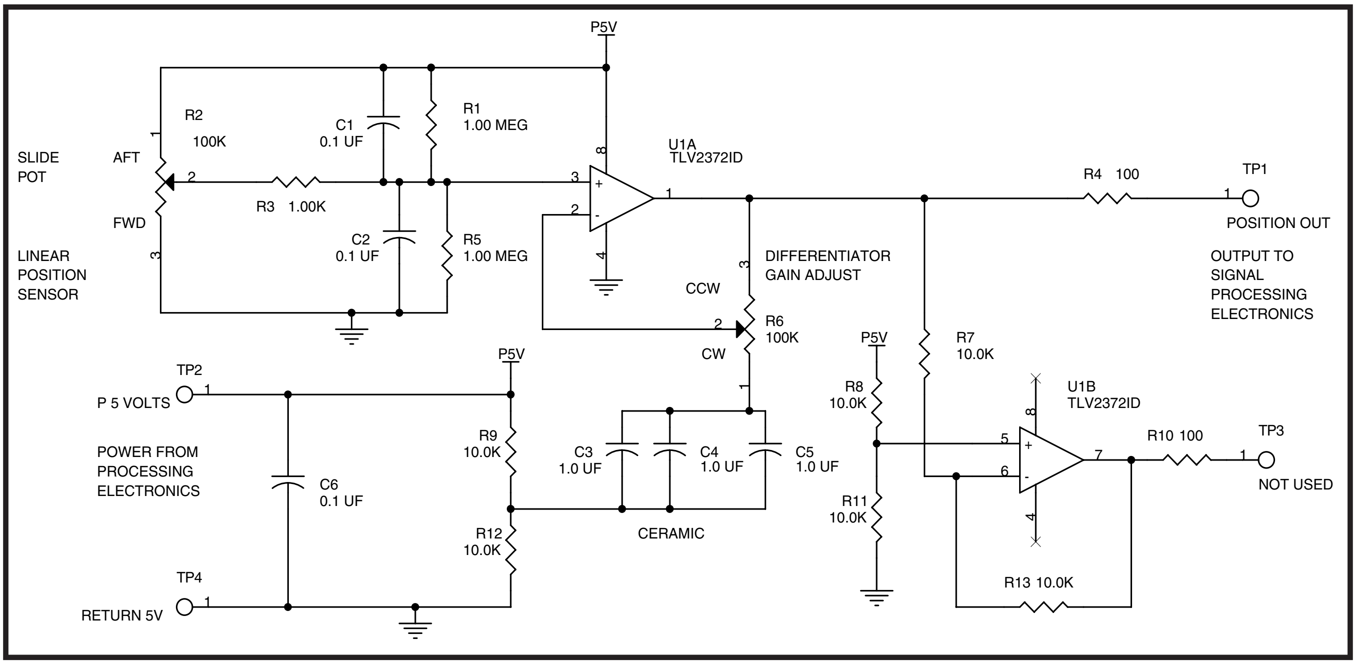
The leading screw-eye forms a pulling handle or a means to hitch the cart to a tractor.

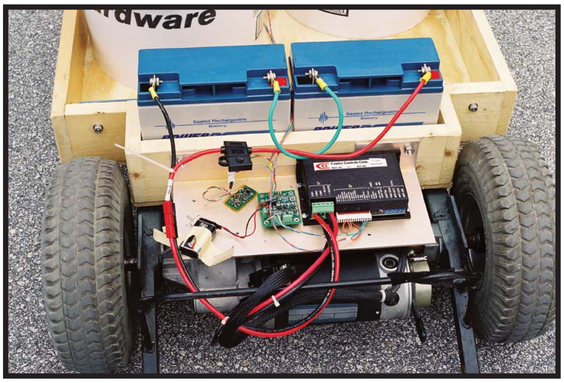
An important feature of the control tongue is that it be made strong enough to pull or push the cart by itself, without the electronic augmentation. This will happen whenever the pushrod reaches one or the other of its mechanical limits of travel. The strength must be in the wood stick and its attachment to the steering post, in the pulling eye and the all-thread, and in the mechanical stops for the all-thread pushrod. The push-wire coupling from the all-thread to the linear position sensor is not part of the strength path. The pushwire is made from a paper clip.

The linear position sensor (or linear displacement sensor) is made from a slider-potentiometer, or slide pot. Very nice linear position sensors are available commercially, each at a price of $250 or more. One commercial example is Penny and Giles SLS095, found through Google on the Internet. A homemade sensor is cheaper and suitable for project development. My assembly is shown in Figure 6. Be careful when selecting slide pots: they are often used in audio applications and so may have an audio taper. For this project, a pot with a linear taper is needed. Also, the length of travel available on the pot is important. The pot that I used has an available travel of about 2.3 inches. A longer pot, with four inches of travel, would be nice. The longer the available travel, the more response lag is allowable. The shorter the pot, the more critical is quicker response from the motor control system. When the pot is short and the lag is long, then you are quite likely to experience saturation of the pushrod travel and the entire pulling or pushing forces of the mass of the cart will be transferred to the agent who is operating the cart. This condition can be handled but it is not the mark of a well-tempered, optimal design.

In my project, I constructed a buffer amplifier circuit board and mounted that near the end of the slide pot. This buffer circuit drives the wires to the signal processor electronics card with a lower source resistance, for lower noise susceptibility. The schematic is shown in Figure 7. The potentiometer, R6, and the associated components C3, C4, and C5 form an adjustable feedback gain for the amplifier U1A. At full CCW rotation, the amplifier provides a straight gain of 1.0 with no frequency shaping in the low-frequency region. At full CW rotation, the AC gain is increased and the amplifier tends to be a differentiator, providing a strong lead term. (This would be the "D" in texts that refer to "PID" systems.) With this lead term, a changing position sensor output controls the motor, so that there is some anticipation, as compared to the operation without the lead term. Variation of the values and settings for C3, C4, C5, and R6 is one of the opportunities for tuning the response of the cart. Note that even for full CW rotation of

FIGURE 7: Electronic schematic of the slide pot buffer amplifier.






■ FIGURE 8: Back end of the cart, showing the electronics assemblies on a mounting shelf.

R6, the DC gain remains at 1.0.

The circuit board is constructed from surface-mount 0805 form factor resistors and capacitors, a dual op-amp — TLV082ID — and a SurfBoard from DigiKey. It was wired with 30 gauge solid copper wire (wire-wrap wire) under a magnifying work lamp. The schematic of the slide pot buffer amp card is shown in Figure 7.

### Signal Processor Electronics

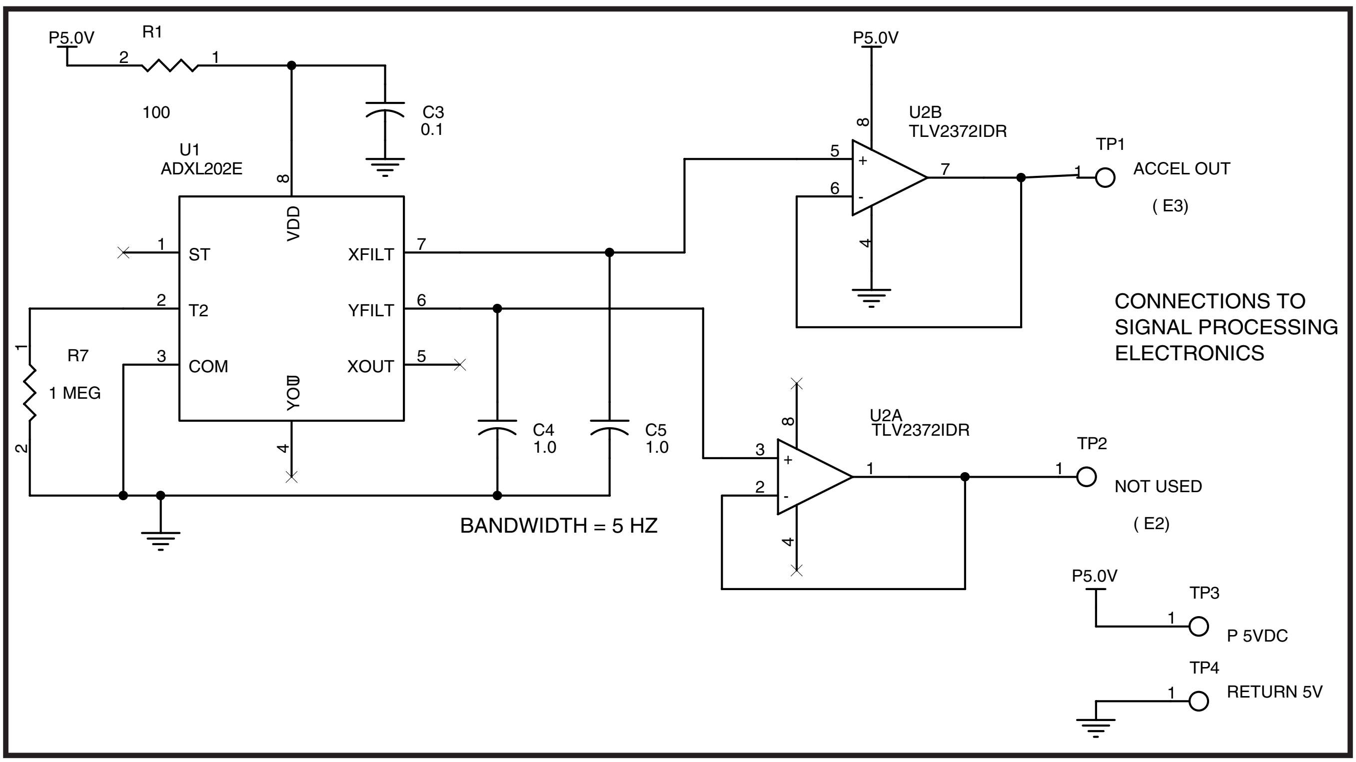
At this stage of the development project, the signal processor electronics are entirely analog — op-amps and a voltage regulator. In some later stage of cart development, I will use a PIC microprocessor and software written in C to implement the gains and frequency compensations, as well as various safety features and non-linear controls. But that is for later. Right now, the name of the game is to achieve the basic functionality and to see if that functionality is useful.

The circuit boards for accelerometer buffer amp and signal processor electronics are shown in Figure 8, along with the Copley Controls power amplifier module. The circuit boards are my designs, reused from previous projects. Their current versions are not suitable for distribution but they are great for my experimentation.

The accelerometer buffer amplifier schematic is shown in Figure 9. An Analog Devices ADXL202 accelerometer is mounted on a prototype circuit board and the analog signal from one of its outputs is wired to a voltage-follower buffer op-amp. (The digital output and the second accelerometer output are ignored.) The schematic is designed in accordance with the ADXL202 data sheet, available from the Analog Devices website (**www.analog.com**).

Physical orientation of the accelerometer relative to the frame of the cart and the polarity of the acceleration signal fed to the gain/summing stage are both very important. The accelerometer input axis must be aligned to fore-aft of the cart. Both the physical orientation of the ADXL202 and the electronic polarity of the gain and buffer stages contribute to the effective polarity of the acceleration feedback signal. The polarity of the acceleration feedback must be correct in order to achieve desired operation. However, the nature of "correct" can be determined in several different ways. As with the polarity of the linear position signal, you can either

■ FIGURE 9: Schematic of the accelerometer circuit card.



be very careful as to details during design or you can work it out at system debug time. Even if you are very careful and very good in your system design and construction, the maximum practically achievable probability of correct polarity connection on the first try is about 55 percent. Or it is 51 percent if you are more normal? You can get 50 percent just by waiting until later and giving it a try. What, me worry?

The signal processor electronics schematic is shown in Figure 10. The basic function of the signal processor electronics is to interface the position sensor and the accelerometer signals to the power amplifier while providing appropriate gain and frequency response for each signal path. All the resistor and capacitor values shown have been determined experimentally.

The signal from the linear position sensor is received and sent to a summing and gain stage, U1A.

The acceleration signal from the accelerometer buffer amp is AC coupled to the U2A stage in order to block the DC offset of the accelerometer output, as well as the continuous component of acceleration due to gravity. The U2B stage is a residual of my prototype board design and it is an opportunity to swap the accelerometer signal polarity. Outputs at TP9 and TP10 go to the differential inputs at the PWM power amplifier. Diodes D1 and D2 provide a small deadzone in the drive to the power amp; this helps to minimize at-rest current drain from the batteries through the amplifier and motor.

The total gain in the path of the accelerometer feedback signal determines the strength of the acceleration effect. Low gains provide less damping of cart position oscillations; too much accelerometer gain tends to make the drive wheels skip around on a concrete floor (with a light load) and the motor operation becomes jerky.

**Power Amplifier**

The power amplifier receives the differential signal from the signal processor electronics and produces a PWM output to drive the brushed DC motor. The present project uses a Model 412, from Copley Controls Corp. (see Figure 8). Details of the amplifier's features may be found at **www.copley.com** This amplifier is designed for general industrial application. It is robust and it is internally constructed so that gains and bandwidths can be tailored to the specific application. It has internal jumpers to select among several different types of feedback control signals, such as voltage feedback, current feedback, and feedback for IR-compensated speed control. There are several different potentiometer adjustments available for input reference (command) signal gain and current-control loop gain. I set the reference signal path gain so that the full output amplitude of the signal processor electronics would cause full rated current from the amplifier.

I configured the amplifier as a straight-gain current-controlled amplifier. I ignored the compensation-configuration opportunities inside the amplifier in favor of my own control circuit boards — this was done only for my convenience. A "final" version would use the configuration features of the amplifier. My connections to the PWM amplifier are in accordance with the information in the manufacturer's datasheet, available online.

A nice feature of the amplifier is an enable/disable input. With a SPST toggle switch wired to this input, the amplifier output can be quickly inactivated when desired, as for a control system fault such as incorrect polarity in a signal path. If the switch is on a long two-wire cable, then you can keep the switch directly in hand while you get familiar with your cart. If those two wires run through a pull-apart connector pair (AMP MTA-100 type) and the amplifier is configured to be disabled for an open circuit, then the cart can disable itself when it unexpectedly tries to run away from you. I know, it happened. Quickly. It is up to you as to what happens when it runs toward you unexpectedly.

**Fuses**

Be sure to include a fuse in the main power path from the battery to the power amplifier. A second, lower-current fuse in the power path to the signal processor electronics and linear position sensor would also be useful. The ground ring on my oscilloscope probe sometimes touches something that it should not have touched.

## Control and Stability Issues

As a servo system, the power-assisted cart is a classic feedback control system that has all the normal opportunities and problems that servo engineers know and love. It has response delays, overshoots, undershoots, and potential oscillations — all the fun stuff. Determining proper gains and time-constants for different elements of the servo loop can be educational, frustrating, and rewarding.

CAUTION: As a motorized servo system employing a powerful motor with a significant gear-reduction and output torque, the cart is also a potential personal safety hazard. If you or any user do something wrong, the results may be undesirable or very undesirable. Building and debugging feedback control systems demands either extreme attention to detail or a good amount of tolerance for things going wrong. A simple reversing of feedback polarity can be catastrophic to something, usually in a mechanical way.

However, determination and insight can produce the magic that is a functioning servo system. It always amazes me when some automatic system that I built does what it is supposed to do, time after time, even when I am not around.

## Controllability Issues

There are several controllability issues that may be addressed. Note that the tongue is normally set up for pulling/pushing with the tongue extending ahead of the cart. The feedback polarity from the tongue sensor is oriented for this use. If the tongue is folded back over the cart, then the sense of elongation/contraction for cart movement is reversed, leading to incorrect polarity of feedback and subsequent run-away operation. This is undesirable. The run-away action could be avoided by limiting the range of elevation of the handle. The tongue foldover could be tolerated if a vertical sensor or tongue-angle sensor were added in order to detect the foldover situation and swap the polarity.

Also note that system position loop gain is nearly zero for the case of the tongue extending nearly perpendicular to the plane of the ground. In this situation, you may not be able to start the cart properly and you cannot stop it satisfactorily.

If the tongue is nearly parallel to the ground but turned to the side as for a turn in the cart travel, then the loop gain is changed and fore/aft surging or oscillation can occur in some conditions. This is one case in which the addition of the accelerometer has been a great help in stabilizing the cart's operation.

## Current Output vs. Voltage Output

The Copley industrial PWM amplifier can be operated either as a voltage output amplifier or as a current output amplifier. Choice of voltage or current output can be a result of careful consideration of servo system performance requirements. Or you can just try both ways. Keeping in mind that the operation of the accelerator pedal in a car is most similar to torque control and thus to current output control, I chose to set up the Copley amplifier as a current/torque controller. Torque control is more likely to result in smooth acceleration and deceleration than is voltage control. However, torque control does not itself result in operation at a fixed speed. With a constant torque setting, speed may increase until something causes an overriding limit. With torque control, there must be a supervisor on hand to adjust torque to obtain desired speed. In your car, that supervisor is you, the driver. You let off on the gas pedal as you get up to about 10 mph over the speed limit. Or you let off on the gas as you come up behind the car in front of you. In the cart project here, the linear position sensor is the supervisor. As cart speed changes, tongue extension changes. When the operator and cart speeds match, there is no change in the separation distance between the two vehicles. Thus, there is no change in tongue extension and torque settles to just the value needed to maintain your speed.

## System Debug Procedures

During system test, lift the rear wheels and block them up off the floor so that the cart cannot run away. Also adjust the reference gain pot on the Copley amplifier for a low gain operation.

Test the signal processor electronics without the accelerometer. Make sure that the linear position sensor controls the motor speed and that the wheels turn in the proper direction.

Turn off power, connect the accelerometer, and turn on power. The linear position sensor must be at its center, zero-signal position. The wheels may turn slightly but should not drive continuously. Lift the front end of the cart a small amount in order to change the orientation of the accelerometer with respect to gravity. The wheels should turn in the forward/reverse direction as you lift the front end of the cart. As you lower the front end back to the floor, the wheels should reverse their rotation. This shows that the accelerometer signal path is working. After you have the front back on the floor, the wheels should come to a stop. The length of time required for the wheels to come to rest is determined by the time constant of the DC blocking capacitors C1 and C2 in the signal processor electronics.

With the cart horizontal and with the drive wheels still off of the floor, slide the cart slightly forward, and then stop; the drive wheels should turn first rearward then forward.

If the polarity of the accelerometer feedback is not correct, you must reverse it. You may do one or more of the following: turn the accelerometer 180 degrees so that its input axis is reversed, reverse the polarity of the buffer amplifier after the accelerometer, reverse the polarity of the inputs to the Copley amplifier, or reverse the connections to the motor. If you choose to reverse the polarity somewhere in the forward path, be sure to make appropriate changes to the polarity of the position sensor signal.

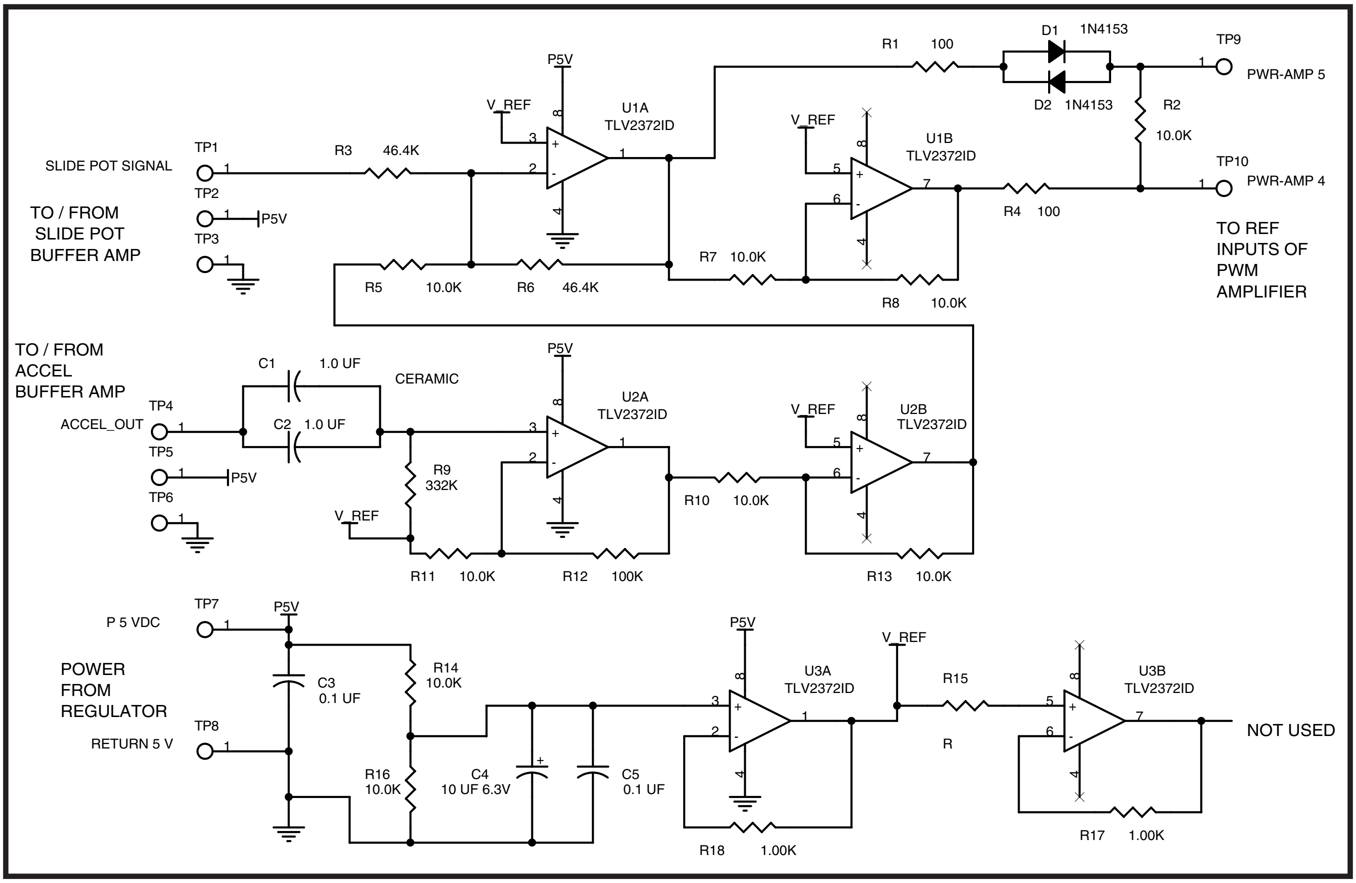
When you think that you have the connections correct, go back and check everything again.

When you are sure that things are correctly connected, turn off power, remove the blocks under the rear end of the cart and set the rear wheels back onto the floor. Have the disable switch in hand — not near at hand, but *in* hand. Then turn on power again. Set the switch to enable the amplifier. Do not touch the ring on the end of the control tongue. Do give the cart a nudge with your foot. If the cart feels heavy and resists moving, the accelerometer is okay. However, if the cart slides away like it was going downhill, then the accelerometer is backwards. It might be wise to consider using the disable switch fairly soon. In this case, reverse something and try again. (Note: the "heavy" cart will decelerate like a heavy cart, as well as accelerating like a heavy cart. Here, "decelerating" means "coasting." Coasting is easily stopped by pushing on the control ring, just as you would for an unpowered heavy cart.)

When the accelerometer feedback is operating correctly, give the ring on the end of the tongue a pull. The cart should move to follow you. Operation is likely to be sluggish since you do have the reference gain turned down at the power amplifier.

If your cart oscillates with fore/aft surges of motion, the gain is too high somewhere. Reduce the gain until the oscillating stops.

Check to make sure that the cart operates better with the power on than with the power off.




■ FIGURE 10: Schematic of the signal processor electronics.

Now you have the basic functionality but have not started the adjustment of gains and frequency response.

## Control System Adjustments

Considerations of gain and frequency shaping can be accomplished in an academic fashion with analysis software such as MATLAB, or they can be adjusted through trial and error. To use MATLAB or any analysis software, you will need a complete knowledge of all the gains and poles and zeros around the two loops: the power amplifier gain (amps per volt), motor torque constant (torque per amp), gear reduction and friction in the differential gear housing, tire diameter, motor armature inertia, cart mass, linear position sensor gain (volts per inch), and accelerometer gain (volts per G or volts per foot per second squared).

In the cart project, the system response parameters that are useful to play with are proportional and differential gain for the position sensor buffer amp, DC blocking time constant and upper frequency rolloff for the accelerometer signal, gain and phase through the signal processor electronics, and gains within the power amplifier.

While MATLAB is an excellent tool, it is most useful if you happen to have it available. If you do not have it available, then the appropriate resistor and capacitor values can be determined by trial and error. It just takes longer and is not as precise as using MATLAB. (I am assuming that you would have used completely accurate data for setting up the MATLAB simulation.)

Your operating cart is essentially an application-specific analog computer. It will tell you what it is doing without any need for a digital computer. A nice thing about electromechanical feedback control systems is that you can see and feel the responses.

Whether you use MATLAB or trial and error, you eventually must go out and try the cart in different situations of loads, ground slopes, ground surface types, and starting and stopping times and travels.

A basic method of testing servo system response is to give the system a step input. This means that you would take the ring in hand and pull it straight ahead quickly so that the push-rod and the position sensor just do not hit the mechanical stops. Hold the ring fixed at the new position. The cart should move promptly by the same amount that the ring moved. If gains and frequency responses are correct, there will only be a little overshoot (once) or a slight undershoot. If the cart does not quite get to where it should, the position sensor gain may not be high enough. If the cart overshoots strongly, then oscillates through two or more cycles, either the position sensor gain is too high or the accelerometer gain is too low. If the motor jerks in its mounting, then maybe the accelerometer path gain is too high.

A procedure for adjusting the gains and time constants experimentally is to work them one at a time. Work one for best response, and then move to the next. See if you can get even better response by adjusting the second feature. Then go to the third feature and adjust it, again trying to get more improvement in overall response. Go around and around the list of adjustments, always trying for improvement. It is useful to keep notes of what you have done and the results. You may want to back up in your progress path at some point.

For my cart, the best setting of the gain/differentiator pot R6 is about the halfway point. Turning toward clockwise increases quickness in response but tends to make the cart unstable with light loads.

Try increasing the reference signal gain in the power amplifier. The cart should become more responsive to changes at the pulling end of the tongue. If the cart oscillates, turn down the gain or readjust pot R6.

Adjustment of the gain in the path from the accelerometer determines how heavy the cart feels. Naturally this effect is more pronounced when the cart is lightly loaded; when the cart is heavily loaded, the motor cannot cause an excess of acceleration and the output of the accelerometer is much less significant.

This advice is only a small fraction of the advice you might possibly need. Find a friend, an engineer, or a professor for more complete advice if you need it. But keep trying.

## Applications

The possible range of applications for the basic control scheme of the power-assisted cart is limited only by one's imagination. For this project, my first idea was for a load-carrying cart hitched to a mobility scooter. My second idea was for a walk-along wagon. And there are a number of other ideas. The carts can be small or large. Longer and wider

### ABOUT THE AUTHOR

Thomas Smith has more than 30 years experience as an electronic engineer designing and building electronics for inertially-stabilized, imaging surveillance systems for military and para-military applications. He has a BSEE degree from the University of Tennessee in Knoxville, obtained in 1970.

He has worked for several large aerospace corporations and for several small ones. At present, he is the owner and sole employee of Fern Creek Electronics, Inc., in Orlando, FL. Recently, he designed, produced, and installed the electronics for the Mako Head horizontal stabilizer device now used in the movie industry.

When not employed at profitable work, he listens to the muses for interesting ideas for motorized servo systems.

carts might be made from the components of a powered wheelchair, where one drive motor is used for each of two drive wheels. Thus, the motors and drive wheels may be spaced apart as you wish. A longer and wider cart could be a towed camera platform, since the accelerometer encourages smooth starts and stops and even speeds.

I can see in my mind a train of power-assisted carts. Such a train of additional carts could carry additional loads without additional pulling-load on the leading tractor. And a train of powered carts would present a higher level of challenge for control and stability analysis. Note that the flow of power-assisted carts in a train over a small rise in the road would be similar to the flow of automobiles through a choke point on a crowded interstate highway. There would be similar bunching up and spacing out.

## Further Work

A platform of this size is large enough to allow the addition of more equipment without crowding the cart or worrying about using expensive miniature components. Further work could add new functionality or could optimize controls and dynamic response for a single power-assisted cart or a train of carts. The control dynamics of a train of power-assisted carts could furnish a challenge for a nearly endless interest in system analysis with MATLAB or its equivalent. One study would be the optimization of system parameters for optimum dynamic response to force perturbations in the middle of the length of the train.

Any use that involves transportation of human people has the added responsibility or liability for safety and or injury; thus, ideas that do not involve human transportation are more interesting to an experimenter.

*Continue to work on the control and stability issues.* Replace the actual tongue and hitch with a virtual hitch. Use acoustic ranging to determine distance. For directional guidance, use an optical target on back of the leader/tractor and track the target. Use wireless networking technology to implement start/stop/speed cooperation among the towed units.

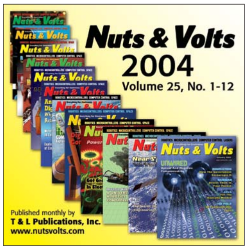
Use weight sensors at the wheels to determine load weight and for improving handling dynamics. Use adaptive signal processing filters to determine load inertia from the immediate history of acceleration versus motor torque.

Add wheel-spin detection, as for loss of traction. **SV**

NEW YEAR'S 10% OFF SALE!
Sale ends 01/31/2006

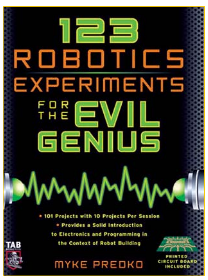
# The *SERVO* Store

## *Nuts & Volts* CD-Rom



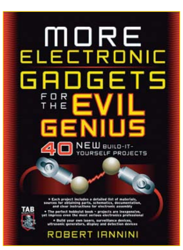

Here's some good news for *Nuts & Volts* readers! Starting with the January 2004 issue of *Nuts & Volts*, all of the issues through the 2004 calendar year are now available on a CD that can be searched, printed, and easily stored. This CD includes all of Volume 25, issues 1-12, for a total of 12 issues. The CD-Rom is PC and Mac compatible. It requires Adobe Acrobat Reader version 6 or above. Adobe Acrobat Reader version 7 is included on the disc. ~~$29.95~~
**SALE PRICE $26.95**

## 123 Robotics Experiments for the Evil Genius
by Myke Predko




If you enjoy tinkering in your workshop and have a fascination for robotics, you'll have hours of fun working through the 123 experiments found in this innovative project book. More than just an enjoyable way to spend time, these exciting experiments also provide a solid grounding in robotics, electronics, and programming. Each experiment builds on the skills acquired in those before it so you develop a hands-on, nuts-and-bolts understanding of robotics — from the ground up. ~~$25.00~~
**SALE PRICE $22.50**

## MORE Electronic Gadgets for the Evil Genius
by Robert E. Iannini

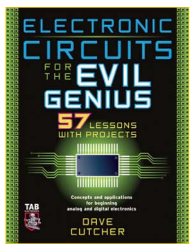



This much anticipated follow-up to the wildly popular cult classic *Electronic Gadgets for the Evil Genius* gives basement experimenters 40 all-new projects to tinker with. Following the tried-and-true Evil Genius Series format, each project includes a detailed list of materials, sources for parts, schematics, documentation, and lots of clear, well-illustrated instructions for easy assembly. Readers will also get a quick briefing on mathematical theory and a simple explanation of operation. ~~$24.95~~
**SALE PRICE $22.45**

## *SERVO* CD-Rom

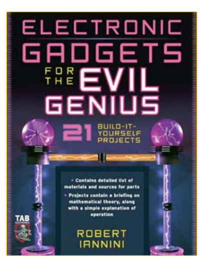

Starting with the first *SERVO* issue — November 2003 — all of the issues through the 2004 calendar year are now available on a CD that can be searched, printed, and easily stored. This CD includes all of Volume 1, issues 11-12 and Volume 2, issues 1-12, for a total of 14 issues. The CD-Rom is PC and Mac compatible. It requires Adobe Acrobat Reader version 6 or above. Adobe Acrobat Reader version 7 is included on the disc. ~~$29.95~~
**SALE PRICE $26.95**

## Electronic Circuits for the Evil Genius
by Dave Cutcher




Cutcher's 57 lessons build on each other and add up to projects that are fun and practical. The reader gains valuable experience in circuit construction and design and in learning to test, modify, and observe results. Bonus website **www.books.mcgraw-hill com/authors/cutcher** provides animations, answers to worksheet problems, links to other resources, WAV files to be used as frequency generators, and freeware to apply your PC as an oscilloscope. ~~$24.95~~
**SALE PRICE $22.45**

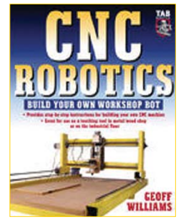
## Electronic Gadgets for the Evil Genius
by Robert Iannini




The do-it-yourself hobbyist market — particularly in the area of electronics — is hotter than ever. This book gives the "evil genius" loads of projects to delve into, from an ultrasonic microphone to a body heat detector, all the way to a *Star Wars* Light Saber. This book makes creating these devices fun, inexpensive, and easy. ~~$24.95~~
**SALE PRICE $22.45**

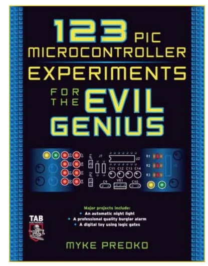
**We accept VISA, MC, AMEX, and DISCOVER**
**Prices do not include shipping and may be subject to change.**

## CNC Robotics
by Geoff Williams



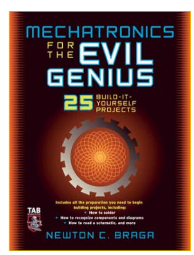

Now, for the first time, you can get complete directions for building a CNC workshop bot for a total cost of around $1,500.00. *CNC Robotics* gives you step-by-step, illustrated directions for designing, constructing, and testing a fully functional CNC robot that saves you 80 percent of the price of an off-the-shelf bot and can be customized to suit your purposes exactly, because you designed it. ~~$34.95~~
**SALE PRICE $31.45**

## 123 PIC Microcontroller Experiments for the Evil Genius
by Myke Predko




Few books take advantage of all the work done by Microchip. *123 PIC Microcontroller Experiments for the Evil Genius* uses the best parts, and does not become dependent on one tool type or version, to accommodate the widest audience possible. Building on the success of *123 Robotics Experiments for the Evil Genius*, as well as the unbelievable sales history of *Programming and Customizing the PIC Microcontroller*, this book will combine the format of the evil genius title with the following of the microcontroller audience for a sure-fire hit. ~~$24.95~~
**SALE PRICE $22.45**

## Mechatronics for the Evil Genius
by Newton C. Braga



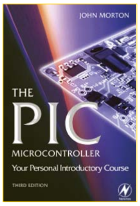

If you're fascinated by electronics and mechanics, this hands-on tour of the junction where they meet will bring you hours of fun and learning. Noted electronics author Newton Braga's *Mechatronics for the Evil Genius* guides you step-by-step through 25 complete, intriguing, yet inexpensive projects developed especially for this book. You will build your own mechanical race car, combat robot, ionic motor, mechatronic head, light beam remote control, and 20 other entertaining learning projects that take you to the heart of mechatronics. ~~$24.95~~
**SALE PRICE $22.45**

**To order call 1-800-783-4624 or go to our website at www.servomagazine.com**

**NEW YEAR'S 10% OFF SALE!** Sale ends 01/31/2006

## The PIC Microcontroller: Your Personal Introductory Course Third Edition

by John Morton

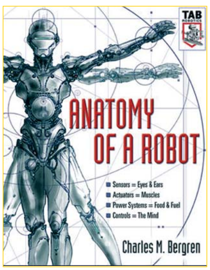



Assuming no prior knowledge of microcontrollers and introducing the PIC microcontroller's capabilities through simple projects, this book is ideal for electronics hobbyists, students, school pupils, and technicians. The step-by-step explanations and the useful projects make it ideal for student and pupil self-study — this is not just a reference book — you start work with the PIC microcontroller straight away. ~~$26.95~~

**SALE PRICE $24.25**

## Anatomy of a Robot

by Charles Bergren

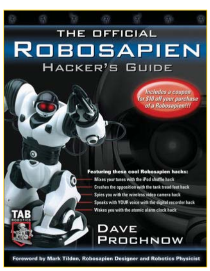



This work looks under the hood of all robotic projects, stimulating teachers, students, and hobbyists to learn more about the gamut of areas associated with control systems and robotics. It offers a unique presentation in providing both theory and philosophy in a technical yet entertaining way. Reading *Anatomy of a Robot* is like having a robot on the operating room table. Crack open the pages, and you'll be able to dissect a robot from head to toe — or from project management and design to motors and power systems. ~~$29.95~~

**SALE PRICE $26.95**

## The Official Robosapien Hacker's Guide

by Dave Prochnow

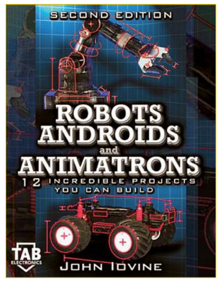



The Robosapien robot was one of the most popular hobbyist gifts of the 2004 holiday season, selling approximately 1.5 million units at major retail outlets. The brief manual accompanying the robot covered only basic movements and maneuvers — the robot's real power and potential remain undiscovered by most owners — until now! This is the official Robosapien guide — endorsed by WowWee (the manufacturer) and Mark Tilden (the designer). This timely book covers possible design additions, programming possibilities, and "hacks" not found any place else. ~~$24.95~~

**SALE PRICE $22.45**

## Robots, Androids, and Animatrons — Second Edition

by John Iovine

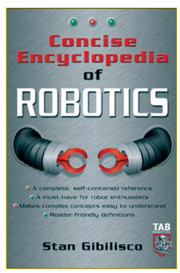



Build your own walking, talking, thinking robot with the projects covered in this book on amateur robotics. Providing you with the building blocks of electronics and motion control, the book enables you to construct a robotic arm that responds to your spoken command, put together basic modules to create sophisticated robot designs of your own, and more. First-time robot-builders and advanced hobbyists can complete these projects without programming or assembly language skills. ~~$19.95~~

**SALE PRICE $17.95**

## Concise Encyclopedia of Robotics

by Stan Gibilisco



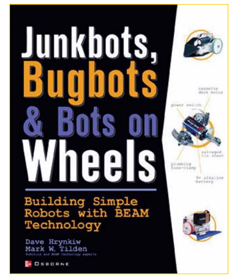

This handy collection of straightforward, to-the-point definitions is exactly what robotics and artificial intelligence hobbyists need to get and stay up to speed with all new terms that have recently emerged in robotics and artificial intelligence. Written by an award-winning electronics author, the *Concise Encyclopedia of Robotics* delivers 400 up-to-date, easy-to-read definitions that make even complex concepts understandable. Over 150 illustrations make the information accessible at a glance and extensive cross-referencing and a comprehensive bibliography facilitate further research. ~~$19.95~~

**SALE PRICE $17.95**

**Check out our online bookstore at www.servomagazine.com for a complete listing of all the books that are available.**

## JunkBots, Bugbots, and Bots on Wheels

by Dave Hrynkiw / Mark W. Tilden

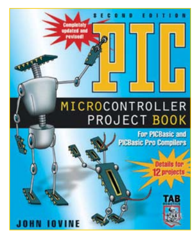



From the publishers of *BattleBots: The Official Guide* comes this do-it-yourself guide to BEAM (Biology, Electronics, Aesthetics, Mechanics) robots. They're cheap, simple, and can be built by beginners in just a few hours, with help from this expert guide complete with full-color photos. Get ready for some dumpster-diving! Get step-by-step instructions from the Junkbot masters for creating simple and fun self-guiding robots safely and easily. ~~$24.95~~

**SALE PRICE $22.45**

## PIC Microcontroller Project Book

by John Iovine




The PIC microcontroller is enormously popular both in the US and abroad. The first edition of this book was a tremendous success because of that. However, many users of the PIC are now comfortable paying the $250.00 price for the Professional version of the PIC Basic. This new edition is fully updated and revised to include detailed directions on using both versions of the microcontroller, with no-nonsense recommendations on which one serves better in different situations. ~~$29.95~~

**SALE PRICE $26.95**

**$20 each 15% off if you buy two or more.**


From HomoSapien to RoboSapien


Before R2D2 there was R1D1

## *SERVO Magazine* T-Shirts

Get your very own limited-edition *SERVO Magazine* T-shirt. Shirts come in sizes S, M, L, and XL, and are available in either black or white. All shirts are 100% pre shrunk cotton.

# EVENTS CALENDAR

Send updates, new listings, corrections, complaints, and suggestions to: **steve@ncc.com** or FAX **972-404-0269**

Another new year has arrived, and it brings with it plenty of robot competitions. Many robot organizations are electing new officers and are just now starting to plan their event calendars. So, we don't have definite dates on many of this year's competitions yet. But there are enough contests already on the calendar to see that it will be another interesting year. The increasing number of outdoor contests means a higher concentration of events in the warmer months.

Several members of the Dallas Personal Robotics Group — where I'm a member — have been busily working on robots designed for the outdoor environment. It's interesting to see the wide range of design choices being implemented. There's a six-wheel skid-steer robot, several robots based on the chassis of the huge 21st Century Toys' Stuart Tank, four-wheel robots based on RC cars, a couple of four-legged walking robots, and one that's similar to the designs used on the Mars rovers. I'm sure members of other robot groups are creating equally interesting outdoor designs. Hopefully we'll be seeing some of these in future issues of *SERVO*.

For the next few months, all of those outdoor robots will have to roll around indoors, but when the weather starts warming up, they'll be going outside. Increasing numbers of strange, outdoor robots will be sighted exploring neighborhoods, parking lots, and university campuses. What could make 2006 seem more like the 21st century than a host of robots moving through everyday, outdoor life?

For last-minute updates and changes, you can always find the most recent version of the Robot Competition FAQ at Robots.net: **http://robots.net/rcfaq.html**

**— R. Steven Rainwater**

## January 2006

**20-26 Techfest**
*Indian Institute of Technology (IIT), Bombay, India*
This is a huge technical festival involving over 15,000 students from 750 colleges across India. There are a lot of other technical contests. Among the robot events this year are Micromouse, G.R.I.P., Full Throttle, AgroBot, and Hover Junior. See the website for descriptions of each event.
**www.techfest.org**

## February

**3-5 Robotix**
*IIT Khargpur, West Bengal, India*
A national-level competition with events such as Mission to Mars, Water Polo, Topsy Turvy, Distance Tracker, and Match Maker.
**http://gymkhana.iitkgp.ac.in/robotix**

## March

**10-11 AMD Jerry Sanders Creative Design Contest**
*University of Illinois at Urbana, Champaign, IL*
This year's contest involves a 44' x 44' course in which robots from multiple teams will navigate ramps, overpasses, and Tee-Tor-Tots in an attempt to collect and dispose of colored balls.
**http://dc.cen.uiuc.edu**

**18-19 Manitoba Robot Games**
*Winnipeg, Manitoba, Canada*
Events may include both Japanese and Western style sumo, mini-tractor pull, and Atomic Hockey.
**www.scmb.mb.ca**

**19-23 APEC Micromouse Contest**
*Dallas, TX*
One of the best-known micromouse competitions in the United States. Expect to see some very advanced and fast micromouse robots.
**www.apec-conf.org**

**21-23 Singapore Robotic Games**
*Republic of Singapore*
Fourteen events including autonomous Sumo, RC Sumo, legged robot marathon, legged robot obstacle course, several levels of micromouse, wall climbers, pole balancers, and more.
**http://guppy.mpe.nus.edu.sg/srg**

## April

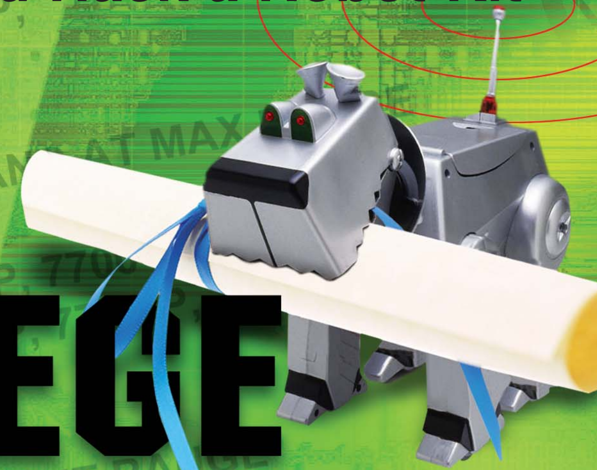
**8-9 Trinity College Fire Fighting Home Robot Contest**
*Trinity College, Hartford, CT*
The well-known championship event for fire fighting robots.
**www.trincoll.edu/events/robot**

**21 Carnegie Mellon Mobot Races**
*CMU, Pittsburgh, PA*
The traditional Mobot slalom and MoboJoust events.
**www.cs.cmu.edu/~mobot**

# KIT KOLLEGE

## A Course on How to Build, Program, and Hack a Robot Kit

### WITH PROFESSOR PROCHNOW



One of the greatest challenges facing robot manufacturers is how to make an introductory kit that is simple enough for beginners, yet sophisticated enough to pique the curiosity of the advanced user. The **Binary Player Robot** by OWI Incorporated is a simple two-wheel robot kit designed specifically for beginning robotics physicists. While this kit is a terrific accomplishment for introducing robot building, it also features a remarkable programming capability that could inspire a whole slew of new advanced robot designs.

Based on a simple BEAM-like transistor-driven design (Biology, Electronics, Aesthetics, Mechanics; reactionary robot designs lacking microprocessor or microcontroller programmed control), Binary Player Robot contains two infrared (IR) sensors which read black and white patterns that are printed on a rotating paper disk. Similar to the IR sensor arrangement used by line-following robot designs, Binary Player Robot instead uses these two IR sensors as a tool for reading this paper disk which functions as the "program" which drives Binary Player Robot.

In order to drive Binary Player Robot, three major subassemblies control this kit's actions:

• *IR Sensors* — Two IR sensors, one assigned to each drive motor. Each sensor reads a specific portion of the program disk and, based on reading either a white or a black pattern, performs one of two basic functions: motor on or motor off.

• *Drive Motors* — Two motors, one driving each wheel. There are three basic settings for each motor: forward/backward/off, three-speeds, and on/off. While the first two settings are controlled by switches installed on Binary Player Robot, each motor can then be independently turned on and off by reading white and black patterns on the program disk. Note that the three-speed settings feature can also be independently set for each drive motor by a shift switch that is built into each motor's gearbox.

• *Program Disk* — A disk-driving motor rotates the program disk over the two IR sensors. The motor spins at a rate of about three revolutions per minute (rpm). These sensors read the black and white patterns on the disk and turn the respective motors off and on. Variations in this programming can enable the robot to move forward, turn, and stop. There are special plastic templates included with Binary Player Robot for creating your own program disks. Several full-size templates have also been included with this lecture.

Class dismissed. **SV**

## THIS MONTH:

**LECTURE 6:**

OWI Binary Player Robot

### HACKS AND MODS

**OWI Binary Player Robot**

- Use diagonal cutters for snipping parts from their plastic trees.
- Identify three motors based on length of connection wire.
- Assembly and greasing of motor gearboxes.
- Installation of wheels.
- Installation of main circuit board.
- Program disk templates.

During the assembly of Binary Player Robot, changes/mods/hacks were employed for streamlining the assembly process, as well as enhancing the final robot's performance. Several full-size program disk templates are furnished in this lecture.


**FIGURE 1.** OWI Binary Player Robot is an introductory robot kit with some sophisticated programming capability. Photo courtesy of OWI Incorporated.

**FIGURE 2.** All parts are bagged and labeled for facilitating quick and easy construction. There are two versions of the Binary Player Robot kit. One requires soldering (OWI-987K) and the other kit does not (OWI-9875). The non-soldering Binary Player Robot kit is the subject of this Kit Kollege.

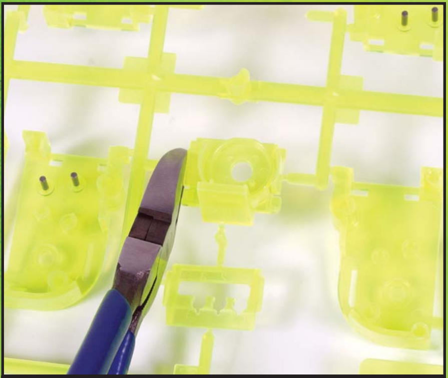


**FIGURE 3.** Use a pair of diagonal cutters for snipping each part from its plastic tree.

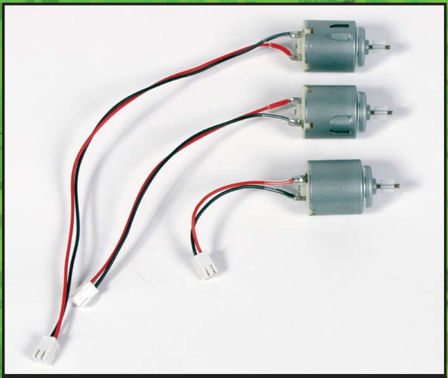


**FIGURE 4.** There are three motors in the Binary Player Robot kit: left wheel drive, right wheel drive, and program disk drive. In order to label the function of each motor, just study the length of each connector cable. The longest cable is for the program disk drive, the medium length cable is the left motor, and the right motor has the shortest cable length.

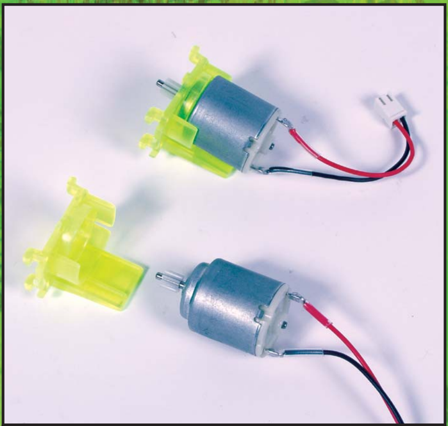


**FIGURE 5.** Mount the left and right drive motors in their respective motor mounting plate. Both mounting plates are the same, there is no assigned left or right plate.

**FIGURE 6.** Build a left and right motor gearbox. The gearboxes are identical until the drive axles are installed.

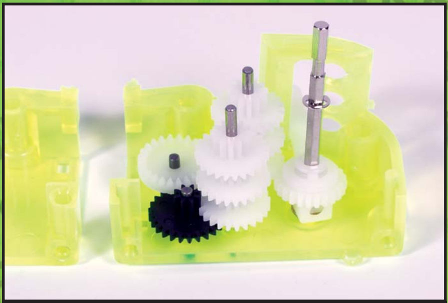


**FIGURE 7.** Pay particular attention to the gear installation and alignment. After the gears are installed, apply a small amount of grease to the gearbox.

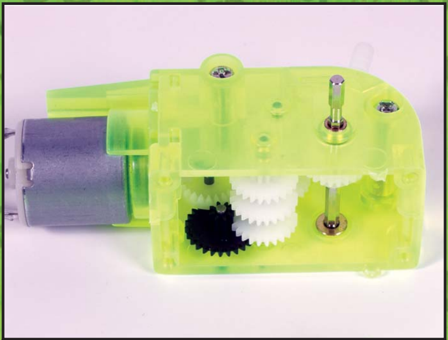


**FIGURE 8.** A completed gearbox includes motor, motor mounting plate, speed shift lever, and drive axle all bolted together.

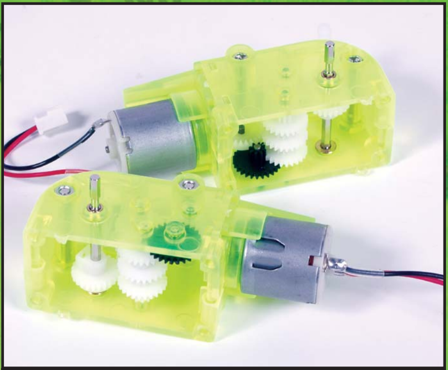


**FIGURE 9.** Both motor gearboxes are assembled with left and right drive axles. Make sure that the right drive axle gearbox is attached to the motor with the shortest connection cable length. Likewise, the left motor gearbox must have the medium length connection cable motor installed. Furthermore, these motors are held tenuously in place and can fall out of the gearbox. Set them aside until needed later.

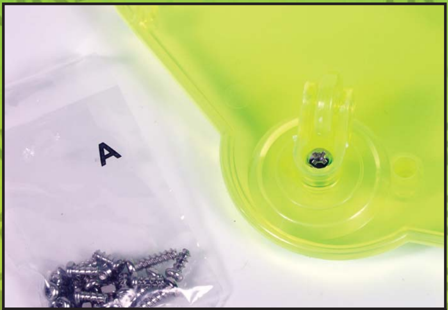


**FIGURE 10.** Mount the front caster wheel to the base plate. Use the hardware from parts bag A.

**FIGURE 11.** Attach two axle set screws to the main drive wheels.

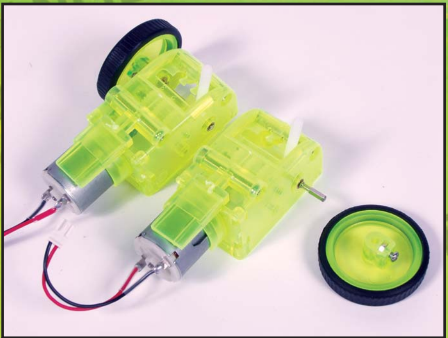


**FIGURE 12.** Slip a rubber tread over each wheel and fix one wheel to each motor gearbox drive axle.

**FIGURE 13.** Drop the large gear program disk holder onto the top of the upper surface of the top cabinet.

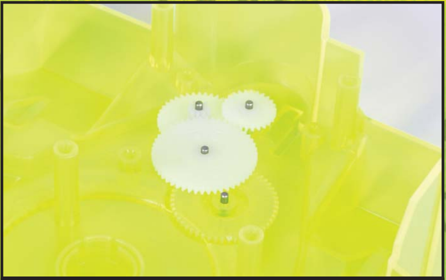


**FIGURE 14.** Flip the top cabinet over and install the program disk gear drive train.

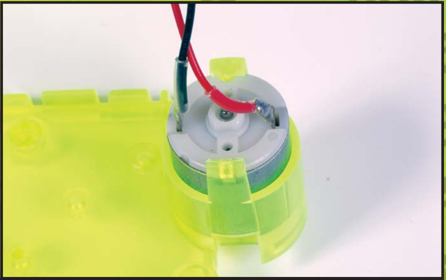


**FIGURE 15.** Firmly seat the program disk drive motor in the motor bracket.

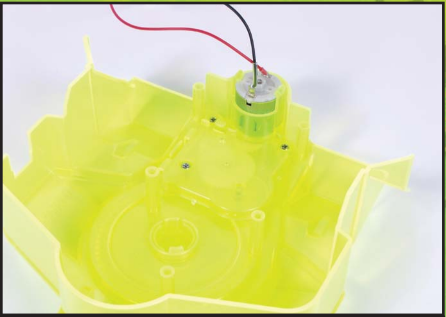


**FIGURE 16.** Mount the motor bracket to the inside surface of the top cabinet.

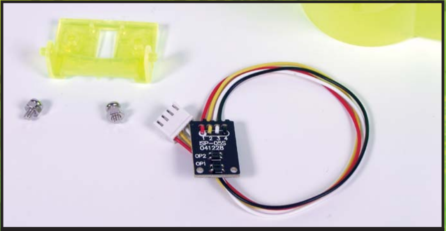


**FIGURE 17.** The program disk is read by two IR sensors which are mounted to a hinged cover that is attached to the top cabinet.

**FIGURE 18.** Install the IR sensors so that they face downward towards Binary Player Robot.

**FIGURE 19.** Snap the IR sensor holder into the hinge along the front of Binary Player Robot.

**FIGURE 20.** A sliding latch acts as a lock knob for holding the IR sensor holder against the program disk.

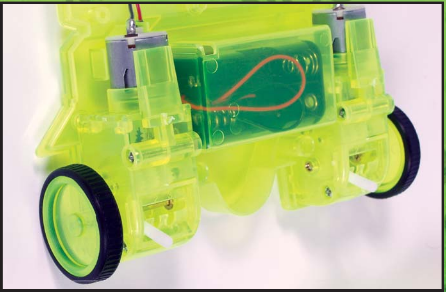


**FIGURE 21.** Mount the two motor gearboxes to the base plate. Make sure that the left and right gearboxes are installed on their correct locations. In this photograph, the left gearbox is at the top of the picture.

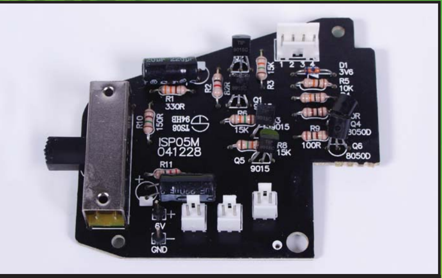


**FIGURE 22.** A main circuit board that any BEAM robotic physicist would love.

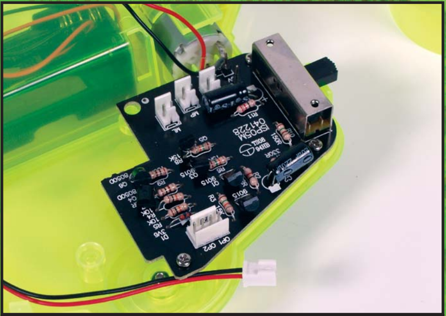


**FIGURE 23.** Mount the main circuit board on the base plate. Be careful not to crimp or pinch the left drive motor connection cable.

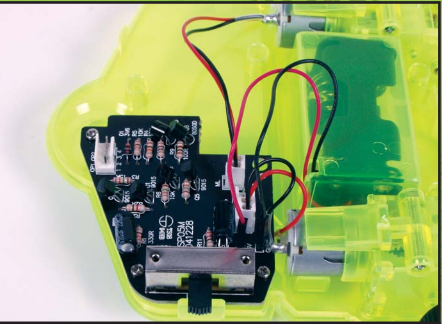


**FIGURE 24.** Attach the two drive motors and battery connections to the main circuit board.

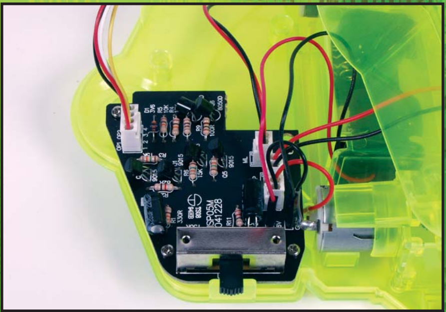


**FIGURE 25.** Connect the IR sensors and program disk drive to the main circuit board.

## BINARY PLAYER ROBOT KIT SOURCES

You can purchase Binary Player Robot from any of the following sources. Please refer to each website for updated pricing information.

*Carl's Electronics*
**www.electronickits.com/robot/owi9875.htm**

*RobotiKits Direct*
**www.robotikitsdirect.com/products/owi9875.html**

*RobotShop*
**www.robotshop.ca/home/products/robot-kits/beginner-robot-kits/owi-binary-player-robot-kit.html**

**FIGURE 26.** The completed Binary Player Robot.

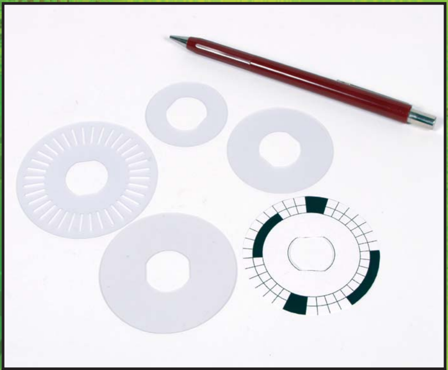


**FIGURE 27.** A set of special plastic templates are included with Binary Player Robot for creating your own "programs." Rather than tracing your own program disks with these templates, a set of full-size templates accompany this lecture.

**FIGURE 28.** A program disk installed in Binary Player Robot. Please note the speed shift levers on the rear of each motor gearbox.

## AUTHOR BIO

*Dave Prochnow is a frequent contributor to Nuts & Volts and SERVO Magazine, as well as the author of 26 nonfiction books including the mega-hit The Official Robosapien Hacker's Guide (McGraw-Hill, 2006) and the upcoming PSP Hacks, Mods, and Expansions (McGraw-Hill, 2006). Dave also won the 2001 Maggie Award for the best "how-to" article in a consumer magazine. You can learn more about this Robosapien book and other robotics/electronics projects at Dave's website:* ***ww.pco2go.com***

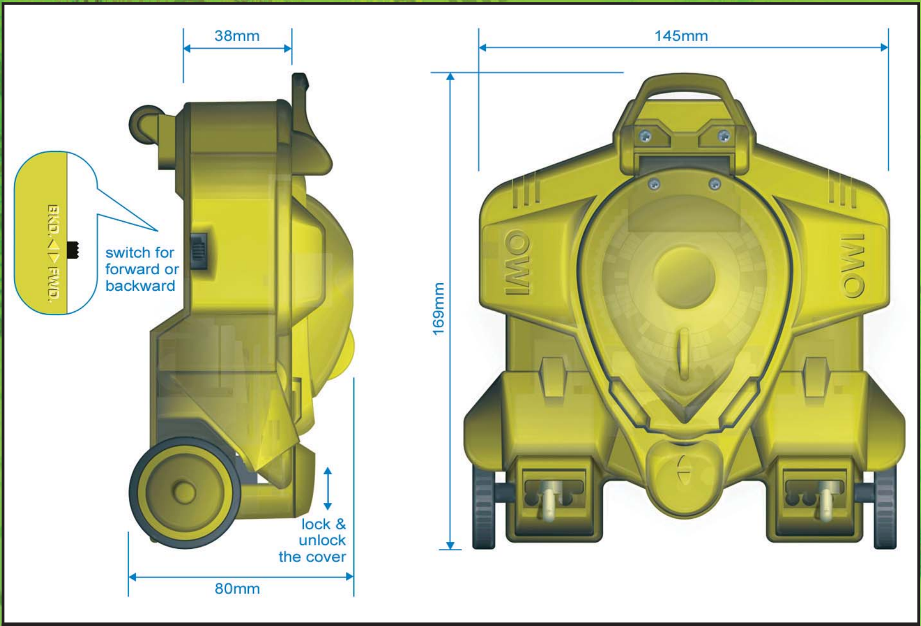



**FIGURE 29.** The large dimensions of Binary Player Robot are ideal for small robotic physicists' hands. Photo courtesy of OWI Incorporated.

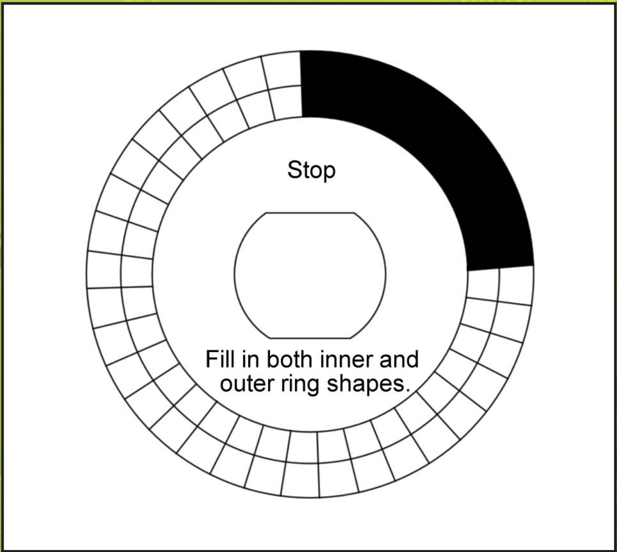



**FIGURE 30.** Stop command. Both rings painted black.

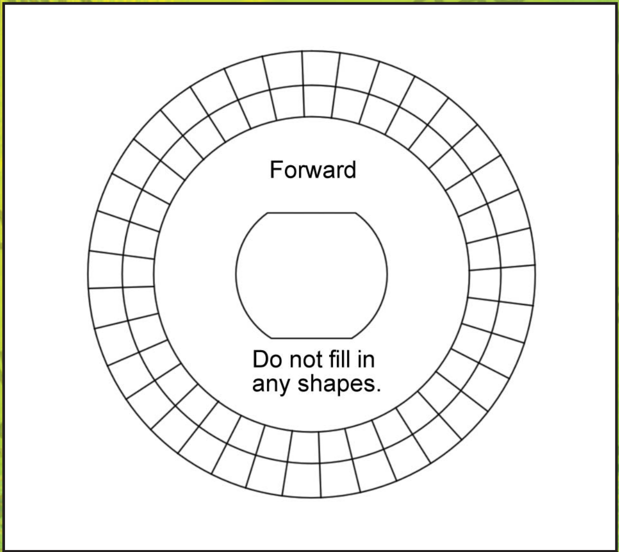



**FIGURE 31.** Forward command. Both rings unpainted.

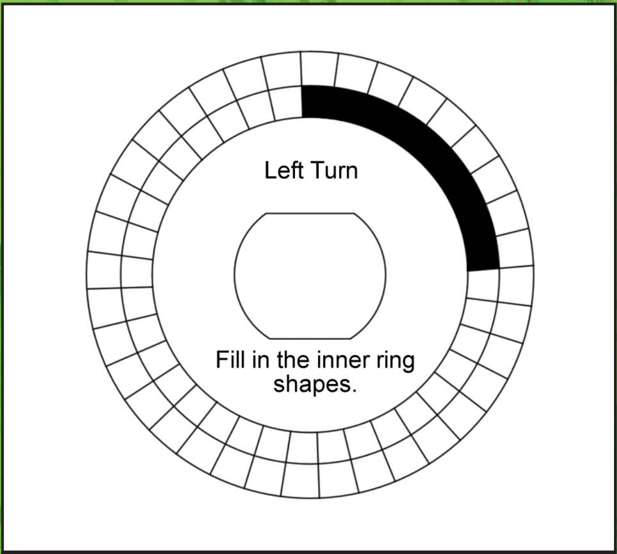



**FIGURE 32.** Left turn command. Inner ring painted black.

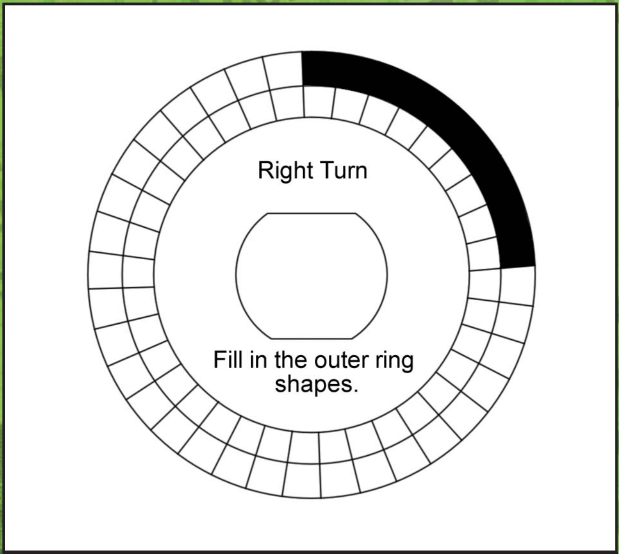



**FIGURE 33.** Right turn command. Outer ring painted black.

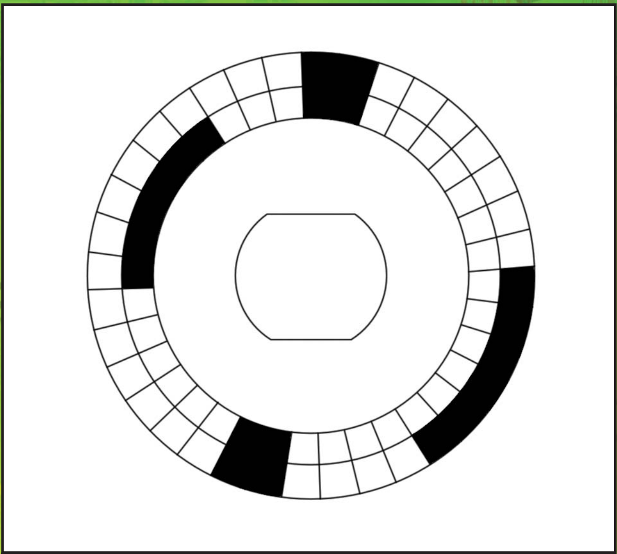


**FIGURE 34.** A complete Binary Player Robot program.

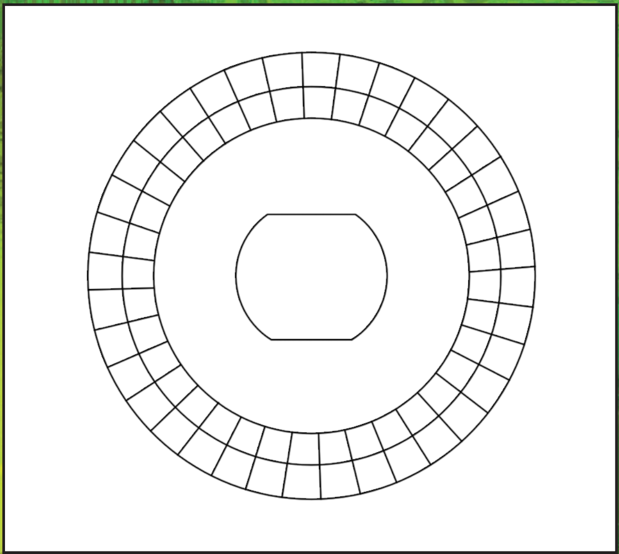
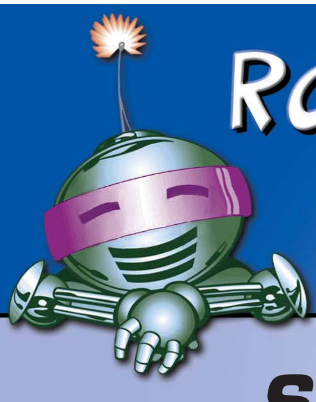


**FIGURE 35.** A blank template.



*Tune in each month for a heads-up on where to get all of your "robotics resources" for the best prices!*

BY GORDON MCCOMB

# So, What Does It Do?

## Developing a Practical Task for Your Robot

From experience, I've found that one of the first things people ask me about my robots is, "So, what does it do?" That's not always an easy question to answer, because the function of a robot can't always be summarized in a quick sentence. Most folks who haven't caught the robot-building bug have neither the patience nor interest to listen to a complex explanation.

Before you can build a robot, you must decide what you want the robot to do. This seems like an obvious step, but the answer is far from obvious. The reason: it's perfectly acceptable to build a robot simply because you want to. In other words, "the journey is the reward," and your reason for building the robot is one of personal satisfaction of going through the process.

Still, many robo-builders prefer to have a definite goal in mind for their creations. It might be a robotics contest pitting the brute-force of two machines against one another. Or, it might be YAVCR — Yet Another Vacuum Cleaner Robot. The goal(s) you set out for your next robot are completely up to you. Defining these goals ahead of time, and keeping them in perspective, will go a long way in guaranteeing a successful project.

### Identifying the Functions of Your Robot

It's always best to view a robot as the machine that it really is, regardless of how smart, cute, or cuddly you think it is. Like any machine, it has a defined set of tasks that it performs. These tasks may be as simple as wandering around your room, or following a black line on a linoleum floor. Or, as complex as navigating your house, sensing the smoke or heat of fire, and vacuuming up dust bunnies that it encounters.

The overall design of your robot will be chiefly dictated by its functionality — a 100 pound stainless steel robot is probably not the best design for a machine that polishes the kitchen floor. The task can be done with a smaller, lighter design, with less risk of damaging walls and furniture.

The first step in establishing a What-it-Can-Do list for your robot is reducing the task (or tasks) you have in mind to a simple list. From there, you can more easily design the size, shape, and capabilities of your robot. Let's create an imaginary homebuilt robot named *Bo buddy*, for "Robotic Buddy," and go through the steps of planning its design. Well start from the standpoint of the jobs it is meant to do.

For the sake of simplicity, we'll design Bo buddy so that it's an "entertainment" 'bot — it's for fun and games, and not built for something like handling radioactive waste, or searching for lost children in building rubble. This is a more important delineation than you may think: a robot designed for its entertainment potential need not be 100% reliable.

The latest industrial and military robots coming off the assembly line are expensive and highly-tuned machines with millions of dollars of research behind them. It can be discouraging to compare your own home-brew creations with these commercial offerings. Count on your robot to make mistakes, and view those mistakes as part of the "entertainment value" of your robot. You'll learn from your robot's mistakes, and you can set about fixing them in your next creation.

### Ye Old Fallback — Vacuum Cleaning

A tried-and-true task for any home-built robot is to attach a 12 volt car vacuum cleaner to it. That way, when some asks you what your latest robot does, you could merely say, "it cleans the floors." That's almost always guaranteed to elicit a positive response. Let's make this the first requirement of Bo buddy: it must be equipped with a vacuum cleaner.

Since Bo buddy is designed to be self-powered from batteries, the vacuum cleaner needs to run under battery power, too. Fortunately, auto parts stores carry a number of 12-volt portable vacuum cleaners from which to choose. You can also find used Black & Decker DustBusters, which run on rechargeable batteries, at flea markets and garage stores.

Bo buddy must have some kind of mobility, so at the very least it can move around and vacuum the floor. There are a number of ways to provide locomotion to a robot. But for the sake of description, let's assume we use the common two-wheel drive approach, consisting of two motorized wheels, counter-balanced by one or two non-powered casters. This is the second basic requirement: Bo buddy must have two drive motors and two wheels for moving across the floor.

Since Bo buddy flits about your house all on its own accord, it has to be able to detect obstacles so it can avoid them. This means the robot must be endowed with some kind of obstacle detection devices. This evaluates to the third basic requirement: Bo buddy must be equipped with passive and active sensors to detect and avoid objects in its path.

Vacuuming the floor and avoiding obstacles requires an extensive degree of intelligence. While some basic circuits could certainly drive the robot around the room in a semi-random way, a more flexible design approach is to use a computer, capable of being programming and re-programmed at will. This computer is connected to the vacuum cleaner, sensors, and drive motors.

Finally, the fourth requirement: Bo buddy must be equipped with a computer to control the robot's actions.

To recap:

1. DC—operated vacuum cleaner
2. Dual motor driven (two drive motors, two wheels, two balancing casters)
3. Obstacle sensors
4. Control computer

The four basic requirements may or may not be the ones you want in your robot. However, you get the idea of how to outline the functions of the robot, and match them with a hardware requirement. These four requirements are merely representative of the basic process you need to consider when implementing a design goal for your robot.

## Additional Functions and Features

Depending on your time, budget, and construction skill, you may wish to endow your robot(s) with a number of other useful features, such as:

- *Variable speed motors* so that your robot can get from room-to-room in a hurry, but slow down when it's around people, pets, and furniture.
- *Set and forget* motor control, so that the "brains on board" controlling your bot needn't spend all its processing power just running the drive motors.
- *Distance sensors* for the drive motors so the robot knows how far it has traveled ("odometry").
- *Infrared and ultrasonic sensors* to keep the robot from hitting things.
- *Contact bumper switches* on the robot so it knows when it's hit something, and to immediately stop.
- *LCD panels, indicator lights, or multi-digit displays* to show current operating status.
- *Tilt switches, gyroscopes, or accelerometers* to indicate when the robot has fallen over, or is about to.
- *Teaching pendant and remote control*: move a joystick to control the drive motors and record basic movements.
- *Sound output* (perhaps combining speech, sound effects, and music).
- *Voice input*, for voice command, voice recognition, and other neat-o things.

Few of these features are directly related to the core task of vacuum cleaning, but they can be used for improving performance, providing a better machine-human interface, and/or increasing the entertainment value of the robot.

Depending on your design decisions, some of the additional features mentioned above may come at no or little cost. For example, if you intend to use a PC motherboard (like a VIA mini-ITX board) for the computer of your robot, along with a copy of the Windows operating system, then basic voice input and output functions are available to you at no cost. (This is also true when using Linux as the operating system, though voice features may not be bundled with the distribution that you get. Be sure to check out Linux Robotics, by friend and fellow robot enthusiast D. Jay Newman, if you're interested in basing your robot on the Linux OS.)

## Filling Out the Details

Once you have the basic set of requirements complete, you are ready to fill out the remaining technical aspects of your robot. You'll consider the additional hardware needed to assemble the completed robot, as well as weight and cost.

Continuing with the Bo buddy example, you'll need the following:

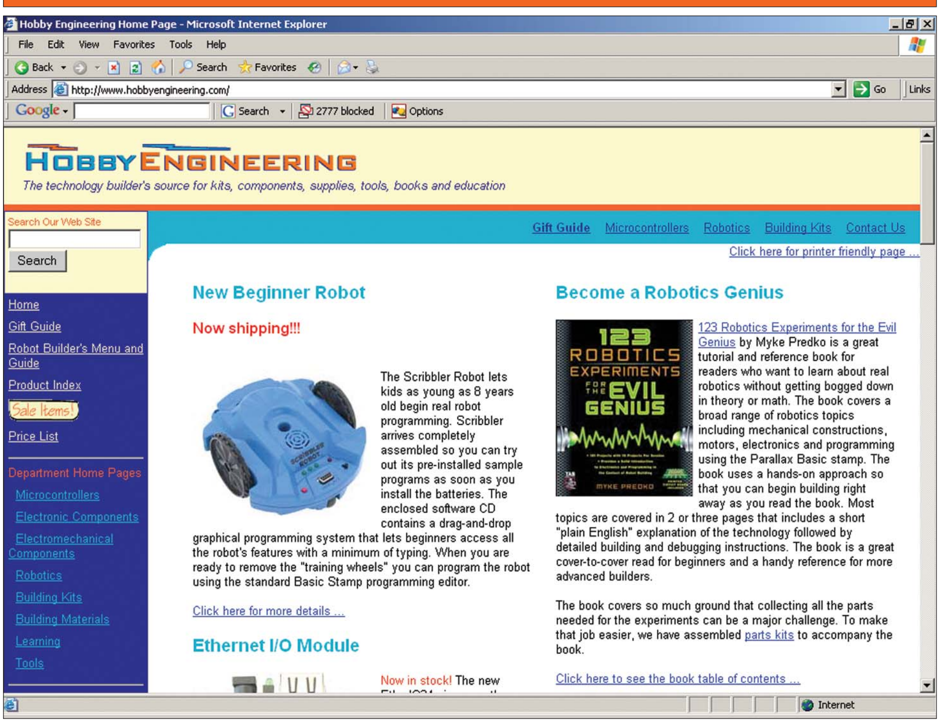
• *Frame or base*. Your robot needs a body for its motors, batteries, computer, and sensors. For works-in-progress, I like the simple round base that you can build up as needed. You can purchase pre-cut bases from any of several robotics supply outlets online, or find sturdy wooden circular bases at the home improvement story. And you can make your own with a jigsaw or router. You can stack additional levels using PVC pipe or extruded aluminum channel stock (also at home improvement stores) as risers.

• *Battery*. By definition, a mobile robot needs its own power source. Battery requirements become quite stringent for a robot outfitted with a vacuum cleaner, and become even more critical if using a PC-based motherboard for a computer. The larger sealed lead-acid batteries are often used for this application. Purchase them by voltage and capacity. Used sealed lead-acid batteries are available surplus, but be sure all the cells in the battery check out.

• *Power supply electronics*. Some of the electronics used in your robot may need voltage regulation. These include the sensors, and possibly the computer board, if it doesn't have its own regulator built in. You will need to purchase or build a regulated power supply according to the specifications of the components you use.

• *Switches, wiring, and other support electronics*. The typical robot is a slurry of fuses, switches, wires, connectors, resistors, capacitors, and other support electronics. You probably won't be able to identify every switch and every wire at the design phase; however, it's a safe bet to make room for these items in your overall design.

**FIGURE 1. HobbyEngineering.com** hosts a bevy of useful kits and parts.




## Avoiding the "*Star Wars* Syndrome"

I call it the "*Star Wars* Syndrome": plans for robots that are simply out of this galaxy. In building robots, it's important to separate the reality from the fantasy.

Fantasy is a *Star Wars* R2-D2 robot projecting a hologram of a beautiful princess. Reality is a home-brew robot that scares the dog as it rolls down the hallway — and probably hits the walls as it goes.

Fantasy is a giant killer robot that walks on two legs and shoots a death ray from a visor in its head. Reality is a foot tall "trash can" robot that pours your house guests a Diet Coke. Okay, so it spills a little every now and then ... now you know why a robot equipped with a vacuum cleaner comes in handy!

It's important to be wary of impossible plans, because they are pretty much a guarantee that your design will fail. When trying to do too much, you run the risk of becoming frustrated with your inability to make the contraption work, and you miss out on an otherwise rewarding endeavor. Don't attempt to give your robot features and capabilities that are beyond your technical expertise, budget, or both. And, let's not also forget the limits of modern science.

When designing your automaton, you may find it helpful to put the notes away, and let them gel in your brain for a week. Quite often, when you review your original design, you will realize that some of the features and capabilities are mere wishful thinking, and beyond the scope of your time, finances, or skills. Make it a point to refine, alter, and adjust the design of the robot before — and even during — construction.

## Sources

The following are general sources for online robot parts.

***Acroname Inc.***
**www.acroname.com**

Acroname is an online retailer specializing in robotics. They carry numerous kits and parts. Also provide a wide selection of sensors: ultrasonic, infrared (including the Sharp infrared distance and proximity), and flame, as well as the OOPic and BASIC Stamp microcontrollers.

***Blue Bell Designs, Inc.***
**www.bluebelldesign.com**

Pre-assembled robotic platform, equipped BASIC Stamp 2p40. No

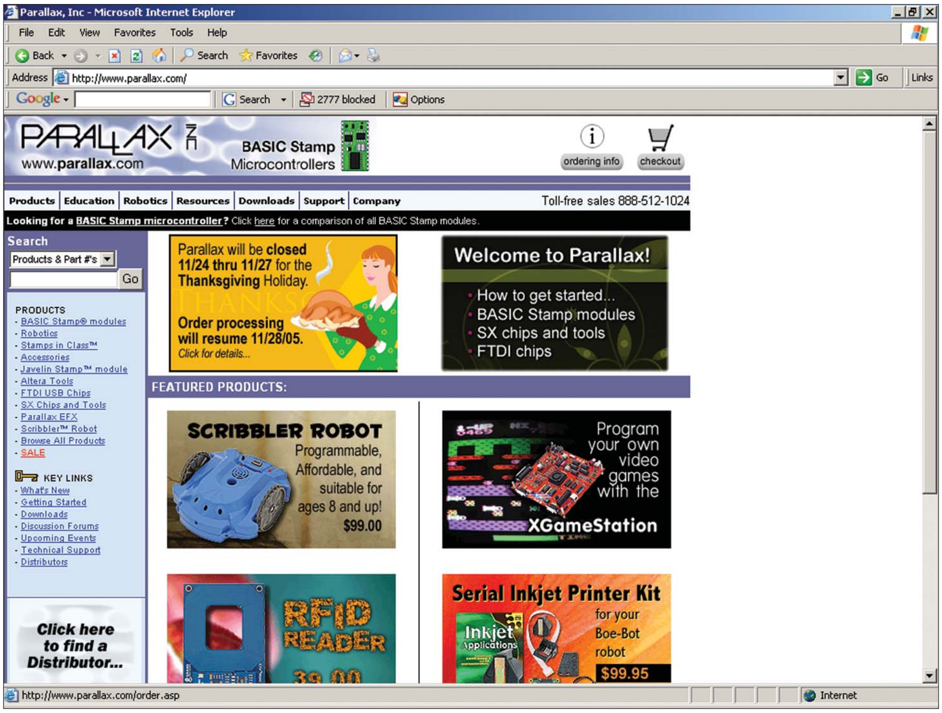



**FIGURE 2.** Parallax is best known for the BASIC Stamp microcontroller, but they also offer a wide range of kits, parts, sensors, and other components for robot building.

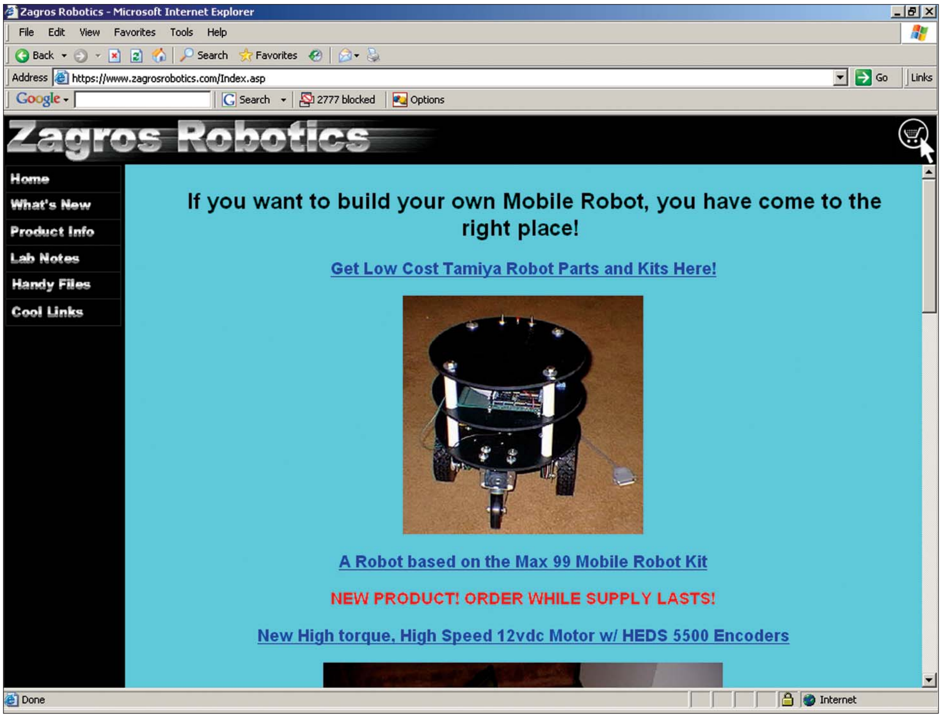



**FIGURE 3.** Zagros Robotics specializes in larger platforms, perfect for that vacuum cleaning bot you've been thinking about building.

assembly required.

### *Budget Robotics*
**www.budgetrobotics.com**

My own company that sells a narrow range of product, with an emphasis on structural elements, such as robot bodies and frames. Much of it is custom-made, and is explicitly designed for affordable amateur robotics. We also buy some hard-to-find items in bulk, and provide them in smaller quantities for your convenience.

### *Hobby Engineering*
**www.hobbyengineering.com**

Full assortment of controllers, sensors, and other components for robotics, as well as bases, starter kits, and construction components.

### *HVW Technologies*
**www.hvwtech.com**

HVW Tech is a leading online Canadian retailer of robotics, microcontrollers, and related products.

### *Images SI*
**www.imagesco.com**

Bot kits, parts, plans, and more.

### *Kadtronics*
**www.kadtronix.com**

Prefabbed robot bases.

### *Kronos Robotics*
**www.kronosrobotics.com**

Kronos sells parts and kits for amateur robotics.

### *Lynxmotion*
**www.lynxmotion.com**

Lynxmotion sells high-quality kits and parts for mobile robots — both wheeled and legged — as well as robotic arm trainers. Check out their latest "Erector Set" style of robot building block components.

### *Mekatronix, Inc.*
**www.mekatronix.com**

Manufacturer of autonomous mobile robots, robot kits, microcontroller kits, and robot accessories, as well as educational materials related to science and robotics.

### *Norland Research*
**www.smallrobot.com**

Among other products, Norland offers a small plastic

(1/8-inch ABS) robot chassis — named S.A.M. — reminiscent of the Topo and Bob robots from the old Androbot company of the mid 1980s.

### Parallax, Inc.
**www.parallax.com**

Wide selection of parts, kits, microcontrollers, and more for small hobby robotics.

### *RB Robotics*
**www.rbrobotics.com**

New owners of the RB5X educational robot. As noted on the website, "The RB5X robot has been produced for over 20 years. With a proven track record and a developed set of teaching aids, it is a great choice for any classroom." Full kits and individual parts available.

### *Robot Store*
**www.robotstore.com**

The Robot Store sells all kinds of robotics goodies, from kits to books to individual parts (like motors, sensors, and wheels). See also **www.jameco.com** — the website for the parent company.

### *Solarbotics Ltd.*
**www.solarbotics.com**

Solarbotics is a primary retailer of BEAM robots, both in kit and ready-made form. BEAM robotics was invented by Mark Tilden in 1989, and following a "simple is better" approach to design. Handy resource for kits and parts for smaller bots.

### *Zagros Robotics*
**www.zagrosrobotics.com**

Zagros offers larger size robotics bases, which include the plastic base plate of the robot and pair of motors. Plus drive electronics, controllers, and more. **SV**


### ABOUT THE AUTHOR

*Gordon McComb is the author of the best-selling Robot Builder's Bonanza, Robot Builder's Sourcebook, and Constructing Robot Bases — all from Tab/McGraw-Hill. In addition to writing books, he operates a small manufacturing company dedicated to low-cost amateur robotics, **www.budgetrobotics.com**. He can be reached at robots@robotoid.com*


# Why Humanoids

## Robots Based on a Humanoid Form Factor Will Eventually Join Us at Home, in the Workplace, and in Public Areas

**by Dan Kara**

I would like you to try the following mind experiment. It is nothing on par with the mind experiments involving Einstein's elevator or Schrödinger's cat, but it does provide some insight into a debate within the robotics community, and a significant debate at that.

To set the stage, picture yourself viewing representations of robots, maybe as photos in an article, on television, as a film clip on the Web, or better yet, a live demonstration. Now ask yourself the following question, and try not to think too hard before you answer. When viewing robots, what type or form factor draws your attention the quickest and holds it the longest? More specifically, what if the choice was between an industrial robot and its humanoid counterpart? Okay. Now let's try a robotic vacuum cleaner and a humanoid. How about a surgical robot and a humanoid, or a lawn mower and a humanoid? You get the point.

I think you will find that there is little disagreement that humanoid robots hold a distinct fascination across all age groups and levels of technical literacy. To put it more plainly ... the humanoid robots win hands-down in popularity contests. I have seen this scenario played out many times at the various robotics events I have attended or produced. Consider the RoboCup championships held in Osaka, Japan in July 2005. The RoboCupRescue and RoboCupJunior events, along with the various RoboCupSoccer competitions, were all well attended by large numbers of enthusiastic fans. However, it was the humanoid RoboCupSoccer events that were the biggest crowd pleasers, even though these events had the least amount of real 'action.'

Other examples abound. Without going into the details just consider R2D2, Honda's Asimo, NASA's Robonaught, or the success of Wow Wee Robotics' Robosapien. What about the DARPA Grand Challenge you might ask? Surely, an event where 23 autonomous robotic vehicles race over a 131-mile desert course for a $2M prize captures the imagination? Undoubtedly. The event received wide press coverage. However, imagine if the Grand Challenge was replaced with a RoboDash event — 10 humanoid robots lined up and competing in a 100 yard dash. Now try to visualize the amount of press that event would receive. It would be staggering. Does the research and results of the Grand Challenge have greater applicability in the real world than our fictional humanoid RoboDash? Sure. Does it matter to the world at large? Not one bit.

While there is no disagreement as to the emotional appeal of humanoid robots, the same cannot be said as to the role of humanoid robots or the humanoid form factor for serious work. The arguments against devoting large amounts of resources, in terms of time, expertise, and money, to the development of humanoid robots, are numerous and varied. Most, however, focus on the fact that the humanoid form factor is not an optimal platform in terms of functionality, performance, and cost.

The technical hurdles to overcome to develop humanoid robots are substantial. Locomotion and power utility come to mind, but they are really only the beginning. We are currently at the stage of robotics development where most products, whether they serve the military, civil, commercial, or consumer markets, are devoted to a single task, or at the very most, serve a limited purpose. The next step is to provide these simple robots with the ability to interoperate with other simple robots in networked teams, to perform more complex tasks such as serving as mobile sensor networks.

Yet, such networked systems will not be released from the lab for years to come. The robotics industry is simply not at the point where autonomous, multitasking functionality can be realistically deployed. You will notice that I have said nothing about natural language interfaces or emotive interactions. Yet this is exactly the type of functionality that the 'humanoid' in humanoid robotics implies.

In broad social terms, the technical objections to the development of humanoid robots are moot. The emotional appeal of the humanoid form factor simply trumps other concerns. Research and resources will continue to be dedicated to humanoid robotics at the university level, while toy and educational robotics producers will grow the market from the 'bottom up' by releasing increasingly more powerful and complex robots. In this way there is a type of pincer movement occurring, where the end result will be functional, autonomous, humanoid robots.

Notice I did not say when these humanoid wonders will enter our lives. Think of humanoid robotics as you would a manned mission to Mars. It will take a great deal of work, money, and time, and in no way will we be compensated equally in terms of spinoff technology. It could also be argued that the money could have been better spent. But will there be a manned mission to Mars? Absolutely.

I understand and agree with most of the arguments against the amount of emphasis placed on humanoid robotics development, especially if it comes at the expense of efforts that will result in real products, solving real problems, in the short term. But there also seems to be an undercurrent of disapproval, often expressed as heated verbal opinions, directed at those who take the humanoid route, especially if those robotic developers come from multinational corporations and make large public displays of their humanoids.

Much of the criticism centers on the large amount of press dedicated to humanoid robots that, speaking frankly, do very little. The high end Japanese robots from the likes of Honda and Toyota run through carefully scripted demonstrations supported by scores of engineers. On more than one occasion I have personally heard such performances compared to Professor Marvel in the *Wizard of OZ* manipulating controls behind a curtain to create a breathtaking display of wizardry for an astounded audience. What appears to be miraculous autonomous behavior is nothing more than preprogrammed sequential actions, supported with a dash of remote control. Brilliantly engineered and beautifully choreographed remote control, but remote control nonetheless.

Such large public displays of humanoid robotics — and just to be clear we are talking about Japanese humanoids here — are dismissed by many as simply a marketing ploy — a high-end, high-touch way of exhibiting engineering excellence. An equal number of people write off the displays as demonstrations of national pride, or at least national ego (Japan is universally acknowledged as the home of the world's most stunning humanoid robots).

Many others believe that the emphasis of humanoid robots in Japan is the direct result of a Japanese national preoccupation with comic books, and one comic book character in particular. That character, perhaps cultural icon is a better term, is Astro Boy, or Mighty Atom as he is known in Japan, a humanoid robot from a cartoon series which first screened in black and white on Japan's Fuji Television network in 1963. According to Toru Takenaka, Honda's chief engineer for the ASIMO project, "I belong to the Atom generation. When I was a child, I loved Atom and Tetsujin 28 (another cartoon robot), and I used to be immersed in the robot world."

The impact of Astro Boy on Japanese roboticists cannot be overstated, nor is it as strange as it seems at first blush. Popular culture has had a significant impact on the robotics world, and that impact is not limited to robotic toys. For example, Helen Granier, iRobot's Co-founder and Chairman of the Board, has often stated that it was R2D2 that first engendered in her an interest in robotics.

So, let me get this straight. Japan's preoccupation with humanoid robots, and the spending of considerable sums developing some of the more famous humanoid robots on the planet, is part of a worldwide marketing and branding campaign (a good business investment) and part fixation with a 40 year old cartoon character (a bad business decision). These robots are funded and developed as national representatives of engineering excellence, even though they have little chance of delivering benefits at the national level and in the near future. Unlike, say, the bullet train, another photogenic engineering feat and source of national pride that actually provides a useful service.

Given the incongruities and inconsistencies listed above, I would say that the emphasis on humanoid robots in Japan and in other quarters is not the result of national pride or homage to a cartoon character taken to an extreme. Nor is it naivety, lack of business acumen, or a 20 year old branding campaign. But what then?

In such situations I find it useful to invoke Occam's razor, the principle attributed to the 14th-century English logician and Franciscan friar, William of Occam, who wisely noted that when one is confronted with multiple predictive theories, one should make no more assumptions than needed. That is, all things being equal, choose simpler explanations over more complicated arguments. In the case of humanoid robotics, there is a straightforward and simple answer to the question "Why humanoids?"

The easy answer is that robots will eventually operate in our home and workplaces, so their form factor must conform to those of humans. But I dismiss this explanation, as well. It is possible, and probably easier, to build robots that can navigate stairs, doorways, and halls, but that are not necessarily humanlike in appearance. No, people will continue to develop humanoid robots because eventually there will be humanoid robots. It is a classie case of circular reasoning, but true nonetheless. **SV**


## AUTHOR BIO

*Dan Kara is President of Robotics Trends, the producer of the RoboBusiness (**www.robobusiness2006.com**) and RoboNexus (**www.robonexus.com**) conferences, and publisher of Robotics Trends (**www.roboticstrends.com**), an online news, information, and analysis portal covering the personal, service, and mobile robotics market. He can be reached at dk@roboticstrends.com*

# Then and NOW

# INDUSTRIAL ROBOTS

by Tom Carroll

Those of us who are interested in experimental robotics pretty much know or think we know what a robot is. In this magazine I've covered some of the more popular experimental/hobbyist robots and even some of the unique robots that travel in space, on battlefields, or under the sea. We must concede, however, when most people of the world think of the term "robot" they envision the big industrial robots in factories welding cars or handling hot items in presses. According to *The Robot Institute of America* in 1979, the definition of a robot is: "A reprogrammable, multifunctional manipulator designed to move materials, parts, tools, or specialized devices through various programmed motions for the performance of a variety of tasks." Even *Webster's Dictionary* in 1993 defines them as "an automatic device that performs functions normally ascribed to humans or a machine in the form of a human." Hmmm, neither definition sounds a bit like an experimenter's robot, does it?

The very first machines that might be called "industrial robots" were used in the old atomic labs to handle radioactive materials and were called "master/slave manipulators" and used mechanical linkages to transfer the motion from an operator to the arm in the hot cell. Since they were just mechanical extensions of a person's hands, they still don't fit the RIA's definition. It was not until *Unimation* came into existence that the first true industrial robots were built.

The story of *Unimation* started back in 1946 when a brilliant engineer by the name of George C. Devol patented a general-purpose playback device that used a magnetic process recorder for controlling machines. That was the same year the 'computer' left the government labs and became a tool for industry.

In 1954, Devol filed for a US patent for a "programmable method for transferring articles between different parts of a factory." In 1956, Devol coined the term *Universal Automation* for his new company. At the suggestion of his wife, Evelyn, he later shortened this to *Unimation*, that became the name of the first robot company.

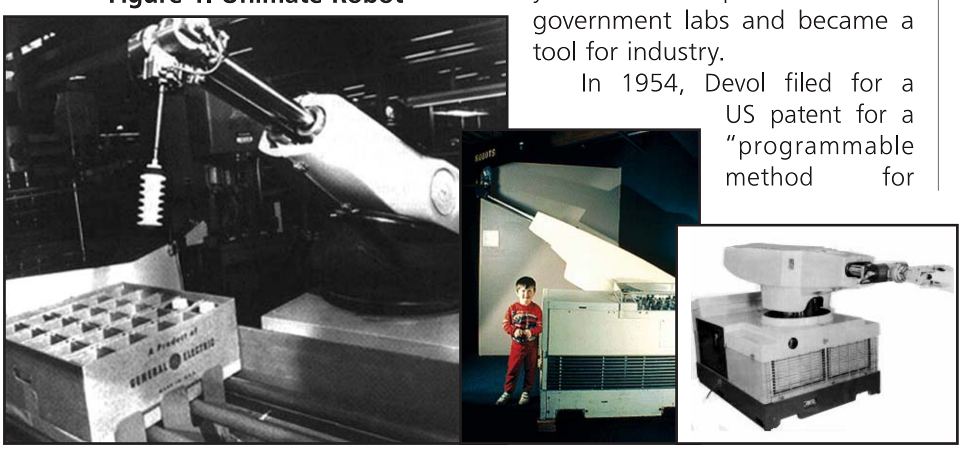
That same year, *Unimation* was to "turn a corner" that would make history. At a cocktail party, Devol met Joseph F. Engelberger — a person who would become the driving force behind this new industry. He was a young engineer in the aerospace industry and during the evening, they exchanged some unique thoughts about factory automation:

- 50 percent of the people who work in factories are really putting and taking (items).

- Why are machines made to produce only specific items?

- How about approaching manufacturing the other way around, by designing machines that could put and take anything.

**Figure 1. Unimate Robot**



Devol and Engelberger built their first machine in 1958. Their machine was a great deal ahead of its time, yet *Unimation* did not show a profit until 1975. Their first industrial robot in 1962, the *Unimate* was placed in a car factory run by General Motors in Trenton, NJ. The robot lifted hot 20 pound pieces of aluminum from a die-casting machine and neatly stacked them — a job deemed too dangerous for a person (Figure 1).

This early machine was hydraulically operated and reminded one of a squat tank turret with a hydraulic piston where one might find a gun barrel. On the end of the piston was the swivel end-effector where the operator could place a gripper or other type of tool. In the days before any type of small computer, the *Unimate* utilized a magnetic drum for its programs and some even used the old "core" type of memory.

Programming was a headache by even the most adept programmer. The operator and maintenance people spent quite a bit of time fixing hydraulic oil leaks and mopping up spilt oil, a design problem that would soon be phased out with newer all-electric robots such as *Unimation*'s popular PUMA series. As many time happens with most small companies, *Unimation* was purchased by AMF and later Westinghouse and finally by the European firm, Stäubli in 1988.

Typical of the way we do things here in the US, Americans are noted for their innovations and the Japanese for their implementation. *Unimation* did not have any patent protection in Japan and their large companies such as Kawasaki Heavy Industries and Hitachi soon began producing superior and less expensive robots than the US and rapidly dominated the market.

In the early 80's, there were close to 2,000 US robot manufacturers, but by the end of that decade one of the last major US robot manufacturers — Cincinnati Milacron — had sold off its robot manufacturing business and European and Japanese companies dominated the business.

**Figure 2. Adept Robot**

Today, there are only a handful of US companies that manufacture industrial robots. There is no comparison of a modern robot to the old robots of the 60's. Adept Technology in San Jose, CA specializes in SCARA robots — those that have all vertical axes of motion (Figure 2).

Yes, the US started it all, but we've been left in the dust by others. The big push now is in mobile service robots, such as the Roomba, lawn mowing robots, health care robots, and military/police robots. Experimenters are the cutting-edge researchers with Robo Magellan and the DARPA Grand Challenge entries that amaze the experts. **SV**

## ABOUT THE AUTHOR

*Retired from Rockwell and NASA projects, Tom Carroll is a space robotics engineer. He's authored numerous articles on combat robots, lives on an island in the Pacific Northwest with his wife, Sue, and enjoys robotics, kayaking, and hiking. He can be reached at: TWCarroll@aol.com*

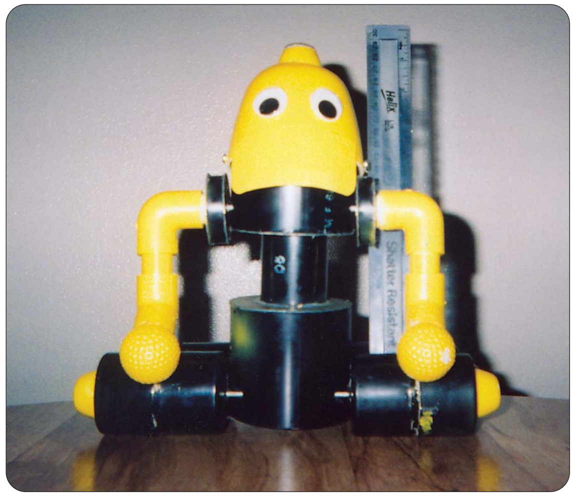
## Advertiser Index

# MENAGERIE

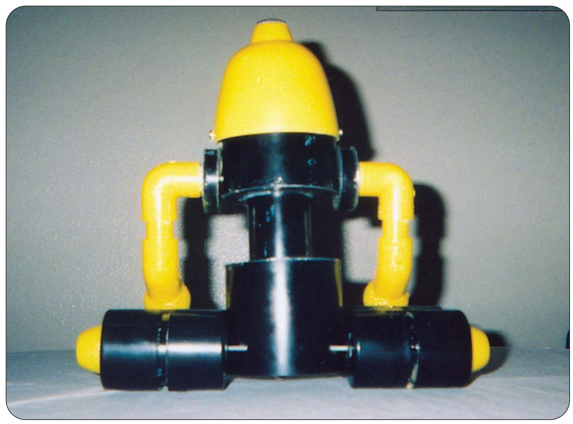
## ROBO BOXERS

*Jerry Baptist*

Robots have become useful in our daily lives, but ROBO BOXERS are for the stimulation of robotic interactive combat sporting activity. ROBO BOXERS box and challenge each other to victory.



ROBO BOXERS
Height: 3 feet 8 inches
Weight: 45 lbs
Rechargeable 12-volt battery
Remote/radio control
Custom metal fabricated and plastic material
Illuminated eyes and contact spots
Timer setting for boxing rounds
Noise activation upon knockout contact

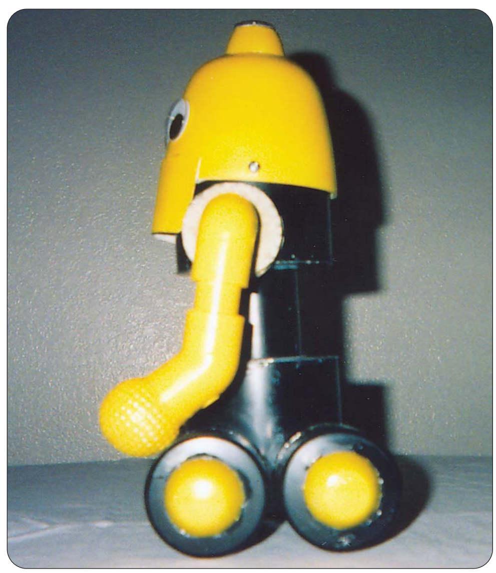


Mini-ROBO BOXERS
Height: 9-1/2 inches
Weight: 8 oz.
Moveable arms/belt and pulley drive system
DC 12-volt battery operation
Mill plastic material

---

*Jerry Baptist developed ROBO BOXERS in 1969. He has been acknowledged for his artistic skills in metal sculpturing. He maintains honorary metal sculptures at the Phoenix Art Museum. He has proficient degrees/certificates in Mechanical Drawing, Electrical Engineering, and Metal Fabrication.*

**www.webspawner.cm/users/jerrylbaptist/**



Send us a high-res picture of your robot with a few descriptive sentences and we'll make you famous. Well, mostly. menagerie@servomagazine.com

# StampWorks

## Micro-Control your World with the NEW StampWorks Experiment Kit

**Our popular *StampWorks* Manual has been revised and rewritten and is back better than ever!** The *StampWorks* manual includes 35 experiments based on the BASIC Stamp® 2 microcontroller and the amazing **Professional Development Board.** StampWorks gives you the hardware, the electrical components and, most importantly, the know-how to become a confident embedded programmer. Working your way through *StampWorks* you will learn about efficient embedded design, connecting circuits and "smart" sensors to the BASIC Stamp, adding computer control to your projects, and "Power PBASIC" programming techniques.

After you've worked your way through *StampWorks* you'll be able to: flash LEDs, use 7-segment and LCD displays, monitor one or several push-buttons or switches, add sounds and sound effects to your projects, build a simple light-controlled theremin, control servos and stepper motors, measure temperature, voltage, and so much more! When you are finished, you will be able to write your own BASIC Stamp programs to control your hobby, engineering, and student projects using a variety of electronic circuits – and you'll be able to do it with the confidence.


| Name | Stock № | Price |
|---|---|---|
| StampWorks Experiment Kit | 27297 | $249.00 |
| StampWorks Manual | 27220 | $24.95 |

**www.parallax.com**

PARALLAX